



## Neurale netværk anvendt til forudsigelse af en geofysisk tidsserie. En sammenligning

Landberg, L.

*Publication date:*  
1992

*Document Version*  
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

*Citation (APA):*  
Landberg, L. (1992). *Neurale netværk anvendt til forudsigelse af en geofysisk tidsserie. En sammenligning*. Denmark. Forskningscenter Risoe. Risoe-R No. 656(DA)

---

### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

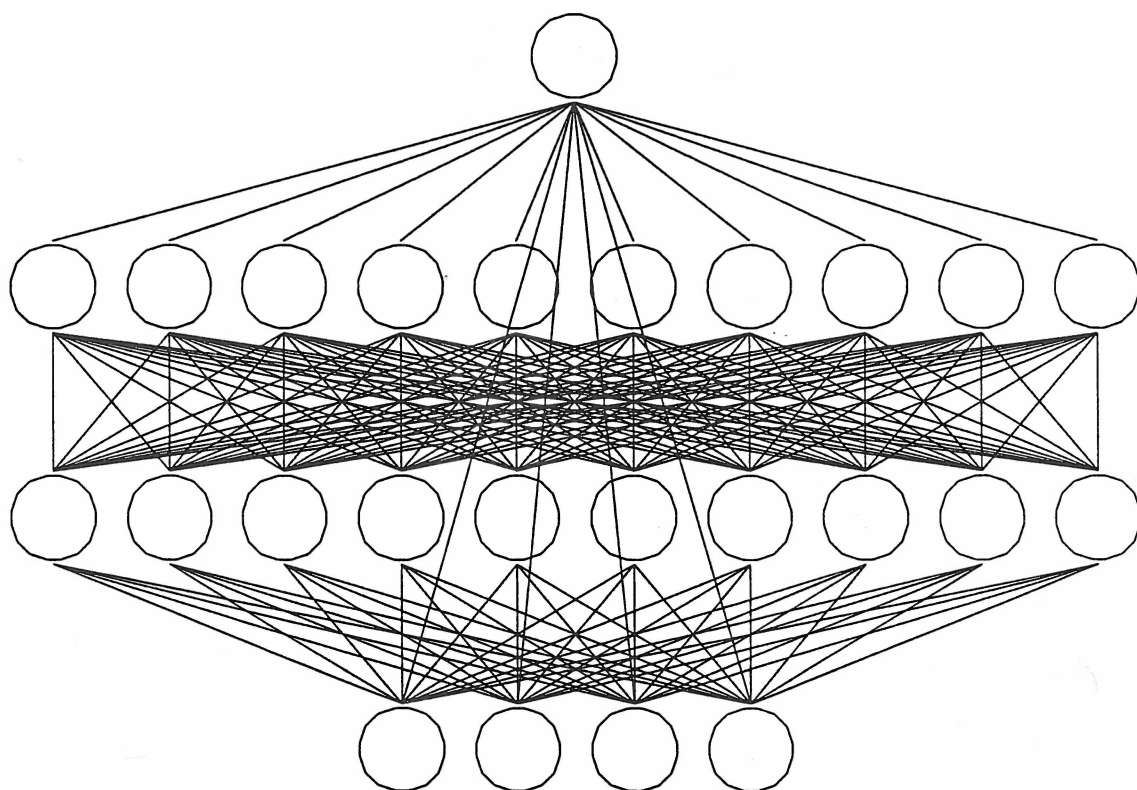
- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

# Neurale netværk anvendt til forudsigelse af en geofysisk tidsserie

## En sammenligning

Lars Landberg



Intentionally left blank

# **Neurale netværk anvendt til forudsigelse af en geofysisk tidsserie**

**Risø-R-656(DA)**

## **En sammenligning**

**Lars Landberg**

**Forskningscenter Risø, Roskilde  
December 1992**



## **Resumé**

I denne rapport beskrives to metoder til at forudsige tidsserier og de evalueres ved hjælp af en temperaturserie fra Fanø. Denne serie dækker en periode på næsten 100 år. Den ene metode bruger neurale netværk (dvs. et stort antal massivt forbundne simple beregningsenheder). Den anden er ARIMA-modellen. Resultatet er at for én dags forudsigelser er neurale netværk, ARIMA-modellen, men også persistens modellen lige gode, men for 5 døgns prognoser er neurale netværk bedre end både ARIMA-modellen og persistens.

ISBN 87-550-1859-9

ISSN 0106-2840

Grafisk Service · Risø · 1992

# Indhold

Forord	4
1 Indledning	5
I      Neurale netværk	7
2 Generelt om neurale netværk	9
2.1 Nerveceller i hjernen	9
2.2 Neuronen	11
2.3 Det neurale netværk	12
3 Delta-reglen	14
4 Indledende eksperimenter med neurale netværk	19
4.1 Generelt om eksperimenterne	19
4.2 Evaluering af resultaterne	21
4.3 Eksperimenterne	22
4.4 Andre netværk til forudsigelse af tidsserier	33
II     ARIMA processen	35
5 Teori	37
5.1 Indledning	37
6 Autokovarians- og autokorrelationskoefficienten	37
6.1 Om temperaturserien	39
6.2 Stationære modeller	41
6.3 Ikke-stationære tidsserier	48
6.4 Modeller for tidsserier med periodiske svingninger	49
7 Modeludvikling	50
7.1 Modelidentifikation	50
7.2 Estimering af parametrene	53
7.3 Diagnostisk undersøgelse af modellen	58
III    Forudsigelse	63
8 Indledning	65
8.1 Forudsigelse ved hjælp af ARIMA-processen	65
8.2 Forudsigelse ved hjælp af neurale netværk	68
8.3 Persistens	71
9 Konklusion	72
Taksigelser	73
Referencer	74

# Forord

Denne rapport er bygget på mit speciale skrevet ved Københavns Universitet, Niels Bohr Institutet og Geofysisk Institut i 1989. Den er dog blevet genlæst og opdateret/rettet på en del punkter.

# 1 Indledning

*Det er svært at spå,  
især om fremtiden.*

*Storm P.*

Formålet med dette projekt er at "spå". Ikke om hvorvidt et menneske får en god eller dårlig fremtid, men blot om det fremtidige forløb af en tidsserie. Til dette formål vil jeg bruge og sammenligne to forskellige "krystalkugler": neurale netværk og ARIMA-processen. Den konkrete tidsserie, jeg vil prøve disse to metoder på, består af målinger af temperaturen på Fanø observeret kl. 8 (Dansk Normal Tid) fra 1. januar 1875 til 9. maj 1969.

Rapporten vil falde i 3 dele: én, hvor teorien for neurale netværk vil blive gennemgået, én, der gennemgår ARIMA (AutoRegressive Integrated Moving Average) modellen og til slut én, hvor begge metoderne bliver anvendt til at forudsige den førnævnte tidsserie og resultaterne sammenlignet.

Grunden til, at jeg har valgt at bruge neurale netværk, er dels at de har vist sig at være *alle* andre forudsigelsesmetoder overlegne, når det gælder forudsigelse af *kaotiske* tidsserier, se Lapedes and Farber (1987), og dels nysgerrighed. For at have en "klassisk" referenceramme har jeg taget ARIMA-modellen med.



---

# **Part I**

## **Neurale netværk**





## 2 Generelt om neurale netværk

Neurale netværk er karakteriseret ved følgende egenskaber:

- *Parallelitet*: Flere beregninger kan foregå samtidigt.
- *Konnektivitet*: Beregningsenhederne er massivt forbundne.
- *Ikke-linearitet*: Beregningsenhederne har en ikke-lineær responsfunktion.
- *Lokalitet*: Beregningsenhedernes lager findes lokalt, da det er de enkelte enheders vægte og tærskler, der er lageret.

Som det ses adskiller et neuralt netværk og en almindelig computer sig på alle de ovennævnte punkter<sup>1</sup>. En computer processerer serielt, det vil sige alt foregår som i en russisk slagterbutik, hvor slagteren (beregningsenheden), farer frem og tilbage mellem disken og lageret for at udføre kundernes ordrer. Dette, at alle beregninger kun kan udføres af én processeringsenhed, resulterer i det, der kaldes *von Neumanns flaskehals*. Et neuralt netværk kan derimod (bedre, men slet ikke dækkende) sammenlignes med et supermarked, hvor kunderne selv udfører deres ordrer<sup>2</sup>.

I det følgende beskrives nogle generelle træk ved neurale netværk. Derefter vil der blive gået mere grundigt tilværks med nogle af emnerne.

### 2.1 Nerveceller i hjernen

Da der i det følgende vil blive talt meget om "neurale" (dvs. neurobiologiske) emner, vil det være på sin plads indledningsvis at gennemgå, hvordan nerveceller eller neuroner, som de også kaldes, er bygget op. For at kunne skelne mellem de neuroner, der indgår i neurale netværk og de, der findes i hjernen, kaldes de sidstnævnte hér udelukkende for nerveceller<sup>3</sup>. Der er cirka 100 milliarder nerveceller i vores hjerne og hvert cellelegeme er nogle få tusindedele mm i diameter. En nervecelle er typisk forbundet med imellem 1,000 og 10,000 andre nerveceller. Den model for nervecellerne, der vil blive gennemgået her, er meget forenklet og svarer til neurobiologiens stadi i 40-50'erne. I de seneste 10-15 år er der sket store fremskridt. Dermed er beskrivelsen af nervecellerne blevet meget mere kompliceret. For en gennemgang af dette se for eksempel Jahnsen og Kiehn (1989). Det gør ikke noget, at de neuroner, der indgår i de neurale netværk, ikke er tro kopier af de nerveceller, der findes i hjernen. Fordi det, det neurale henviser til, er de generelle træk ved nerveceller og ikke en eksakt fysisk lighed.

I Figur 1 ses en nervecelle.

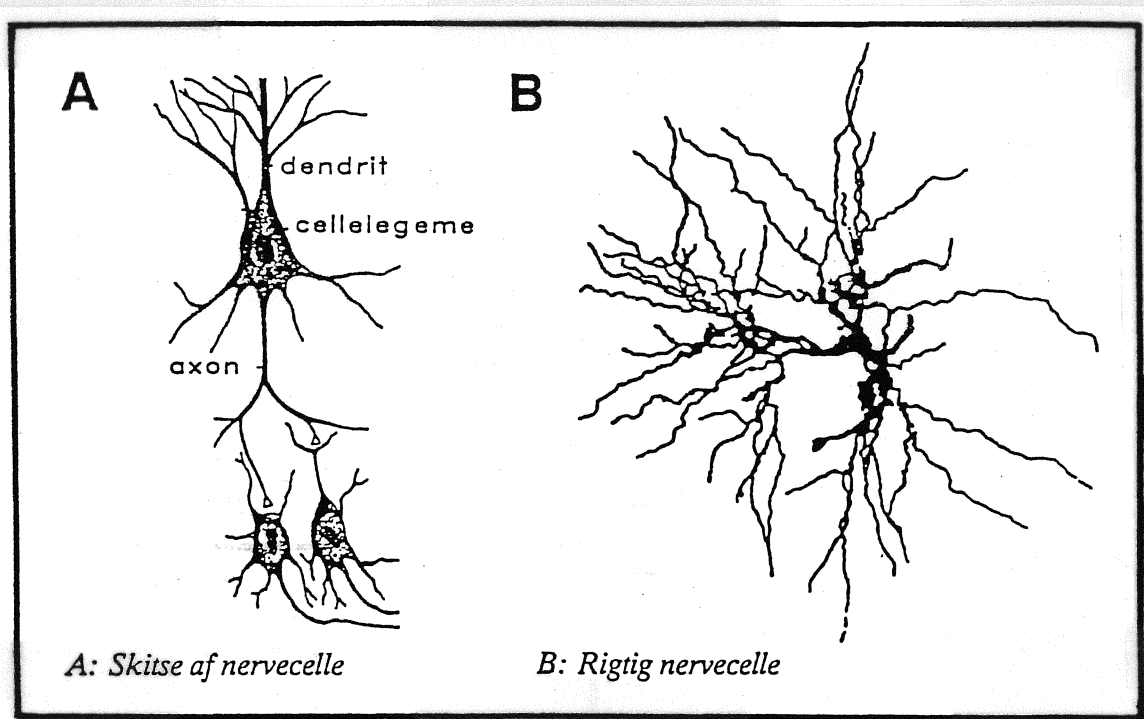
På Figur 1A er vist en skematisk fremstilling af en nervecelle og på Figur 1B en rigtig, det vil sige et fotografi af en kunstigt farvet nervecelle fra den menneskelige hjerne. En nervecelle består af et *cellelegeme* og to slags udløbere: *dendritter* og *axoner*. Dendritterne modtager påvirkninger fra andre nerveceller. Disse påvirkninger kan enten være hæmmende eller fremmende. Modtager nervecellen tilstrækkelig mange fremmende påvirkninger, opstår en *nerveimpuls*, der sendes ud via axonerne. Enden af axonet er ofte forgrenet og ligger tæt ved andre nervecellers dendritter. De enkelte ender af axonet kaldes *synapser*. Når nerveimpulsen når hertil udsondres et signalstof. Man siger, at nervecellen "fyrrer". Signalstoffet

<sup>1</sup>Noget helt andet er, at når man simulerer neurale netværk, så foregår det på almindelige computere. Der er dog ved at være computere fremme på markedet, der kan processere parallelt, for eksempel "The Connection Machine".

<sup>2</sup>Analogierne er hugget fra Brunak og Lautrup's bog (Brunak og Lautrup, 1988)

<sup>3</sup>Andre kalder neuronerne i de neurale netværk for formelle neuroner (Lautrup, 1989) eller model neuroner (Krogh, 1989).





Figur 1. Skematisk fremstilling af den klassiske opfattelse af en nervecelle. A er en skematisk fremstilling og B en rigtig, det vil sige et fotografi af en kunstigt farvet nervecelle fra den menneskelige hjerne.

kaldes en *transmitter*. Det er gennem disse, at én nervecelle modtager signaler fra en anden. Det er disse signaler, der enten hæmmer eller fremmer aktiviteten i den modtagende nervecelle. Der er altså ikke tale om en *direkte* forbindelse mellem nerveceller. Dette er en af årsagerne til, at kommunikationen i hjernen foregår så relativt langsomt som den gør-i forhold til digitale computere. Udbredelse-shastigheden for nerveimpulser er cirka 100 m/s, set i forhold til computerens 300,000 km/s. På grund af hjernens store konnektivitet, er den alligevel den digitale computer langt overlegen med hensyn til udførte operationer per tidsenhed.

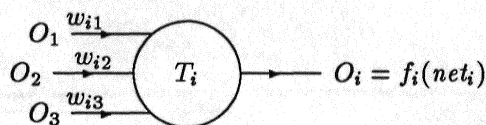
Når en nervecelle er i hvile, er der en spændingsforskel mellem indersiden og ydersiden af cellemembranen. Denne spændingsforskel opretholdes, fordi der i membranen er anbragt "pumper", der aktivt pumper positive ladninger, i form af  $\text{Na}^+$ -ioner, ud af cellen. Der bliver altså et underskud af positive ladninger i cellen i forhold til omgivelserne. Nerveimpulsen – *aktionspotential* – opstår, når der åbnes nogle kanaler i membranen, som kortvarigt tillader passage af  $\text{Na}^+$ -ioner ind i cellen. Kanalerne, der tillader passage af  $\text{Na}^+$ -ioner, åbner, hvis nervecellen bliver mindre negativ indvendigt, end den er i hvile. Den spændingsforskel, ved hvilken der dannes et aktionspotential, kaldes *tærsklen*.

Transmittere kan påvirke nerveceller ved at åbne eller lukke ionkanalerne i membranen. Transmittere, der tillader passage af *positive* ioner ind i cellen, kaldes *ekcitoriske*. Omvendt kaldes transmittere, der lader *negative* ioner passere ind i cellen for *inhibitoriske*.

På grund af den beregningsmæssige analogi, lånes nogle termer fra den virkelige (neurale) verden, når man taler om neurale netværk. For eksempel taler man om at nettet "lærer", "optrænes" og så videre, som var det et menneske eller i hvert tilfælde en bevidsthed, der var tale om. Bemærk dog, at nettene bliver kaldt 1a, 1b og så videre og ikke for eksempel R2D2 eller Fido!

## 2.2 Neuronen

Et neuralt netværk består af en mængde af beregningsenheder. Disse kaldes for *neuroner*, fordi deres måde at arbejde på minder om de nerveceller (neuroner), man finder i hjernen. Dette er også årsagen til navnet neurale netværk. Der er ingen central "kommandopost", der bestemmer over disse neuroner. Enhver neuron ved, hvad den skal gøre og gør det i det øjeblik, den bliver påvirket af et input. På Figur 2 kan man se et eksempel på en neuron. Det ses, at en neuron kan have et eller flere input og ét output. Der gælder generelt, at en neuron kun har ét output - et *neuralt netværk* kan dog have flere output-neuroner og dermed flere output.



Figur 2. Eksempel på en neuron. Denne neuron modtager input fra 3 andre neuroner og giver ét output.

### Typen af de enkelte neuroner.

Det, der karakteriserer en neuron, er på hvilken måde den behandler input, samt hvordan inputtet omsættes til output.

De enkelte input vægtes med en vægtfaktor, de vægtede input adderes og fra denne sum fratrækkes en værdi. For ikke-lineære neuroner kaldes denne værdi for *tærsklen*, på grund af ligheden med nerveceller (se senere og afsnittet om nerveceller side 9). Denne nomenklatur anvendes helt generelt, altså også for lineære neuroner. Det færdigbehandlede input,  $net_i$ , til den  $i$ 'te neuron skrives:

$$net_i = \sum_{j=1}^n V_{ij} O_j - T_i,$$

$V_{ij}$  er vægtfaktoren hørende til det input, der kommer fra den  $j$ 'te neuron<sup>4</sup>,  $O_j$  er outputtet fra den  $j$ 'te neuron,  $n$  er antallet af neuroner som den  $i$ 'te neuron modtager input fra og  $T_i$  er den  $i$ 'te neurons tærskelværdi. Da det er størrelsen af vægtene og tærsklerne, der ændres under oplæringen af et neuralt netværk og dermed dér hvor al informationen opbevares, kan de betragtes som en slags *lokal hukommelse*.

Outputtet fra den  $i$ 'te neuron,  $O_i$ , er en funktion af  $net_i$ . Denne funktion kaldes *aktiveringsfunktionen* eller *outputfunktionen* og den skrives:

$$O_i = f_i\left(\sum_{j=1}^N V_{ij} O_j - T_i\right)$$

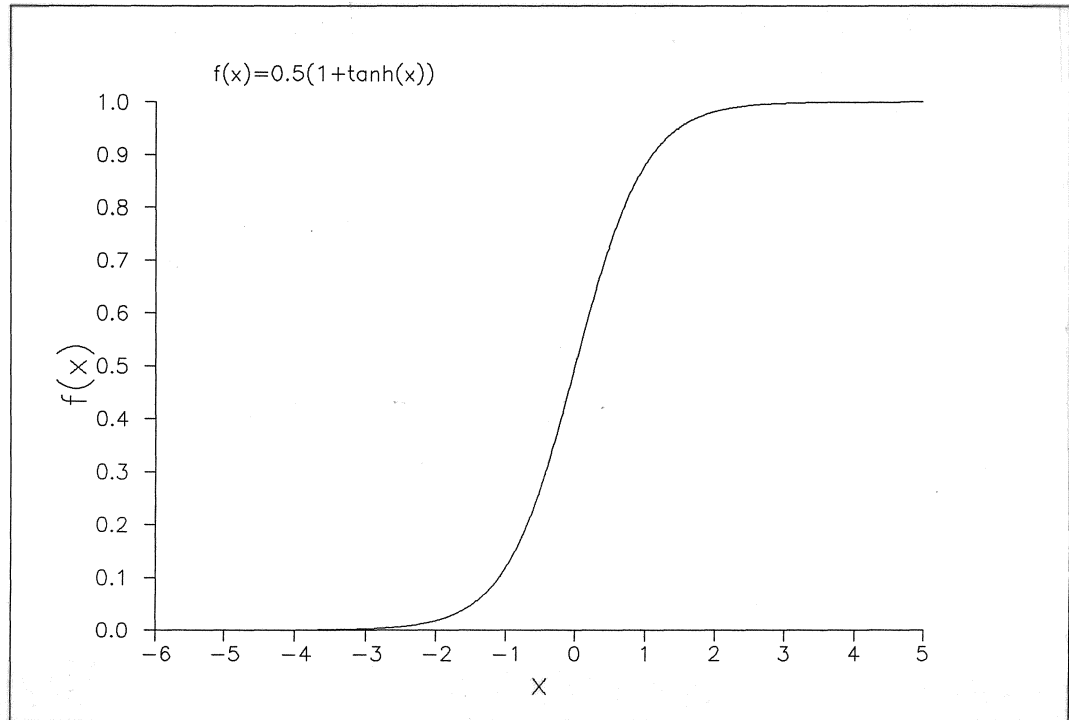
### Aktiveringsfunktionen for de enkelte neuroner.

Alle neuroner har en aktiveringsfunktion. Den mest simple er  $f(x) = x$ , det vil sige  $O_i = net_i$ . Neuroner med denne aktiveringsfunktion kaldes for *lineære neuroner*. Det er nu muligt at drage paralleller til virkelige nerveceller (altså dem, der befinder sig i hjernen). Her vil en *negativ* vægt svare til, at den modtagende neurons output *hæmmes*, det vil sige bliver mindre. Tilsvarende vil en *positiv* vægt

<sup>4</sup>Bemærk, at man i hjerneanalogien kan betragte det som værende forbindelsen mellem neuron  $j$  og neuron  $i$ , der foretager vægtningen, analogt med axonerne/dendritterne (se Figur 1).

bevirke en *forstærkning* af den modtagende neurons output. Neuroner, der befinder sig i det neurale netværks outputlag (se senere), er typisk af denne lineære type.

En anden type af aktiveringsfunktioner er de *semi-lineære* funktioner. Disse kaldes også for sigmoide<sup>5</sup> funktioner, se Figur 3. Af figuren ses det også, hvorfor denne aktiveringsfunktion kaldes for semi-lineær, for den er tilnærmelsesvis lineær i et interval symmetrisk om nul.



Figur 3. Den sigmoide aktiveringsfunktion. Tærsklen er i 0.

Det er altså kun, hvis det vægtede input er positivt eller svagt negativt, at neuronen udsender en impuls. Har en neuron en aktiveringsfunktion som vist på Figur 3, vil den ikke udsende noget signal for input mindre end  $-3$ , omkring  $-3$  kommer der et svagt signal, der vokser til fuldt udslag for input omkring  $+3$  og større. Dette giver forklaringen på, hvorfor  $T_i$  kaldes tærsklen for den  $i$ 'te neuron. Det er netop denne værdi, der – i forhold til inputtet – bestemmer hvilke værdier, der er tilstrækkelige for at få neuronene til at "fyre", det vil sige give output forskelligt fra nul. I denne rapport anvendes funktionen

$$f(x) = 0.5(1 + \tanh x)$$

som aktiveringsfunktion for semi-lineære neuroner.

## 2.3 Det neurale netværk

Neurale netværk er neuroner forbundet med hinanden, således at de danner et netværk. Det, der adskiller det ene netværk fra det andet, er:

- Arkitekturen
- Signalpropageringsmåden
- Oplæringsalgoritmen

---

<sup>5</sup>Sigmoid betyder S-formet.

Et netværks **arkitektur** bestemmer hvilke neuroner, der er forbundet med hvem. Typisk deler man netværket op i lag af neuroner. Der findes tre forskellige slags lag:

1. Input lag
2. Skjulte lag
3. Output lag

Neuroner i inputlaget kaldes inputneuroner, neuroner i det skjulte lag kaldes for skjulte neuroner og neuroner i outputlaget kaldes outputneuroner. Det er inputneuronerne, der modtager input fra omgivelserne. Man kan ikke sige, at inputneuroner har en aktiveringsfunktion, for de sender input'et ubearbejdet videre. Skal man forsøge at tildele dem en aktiveringsfunktion, bliver den af den lineære type, med fast vægt lig 1, og fast tærskel lig 0 (nul). Input'et, som inputneuronen modtager, kan være af mange forskellige slags. For eksempel kan det være signaler fra lysfølsomme celler, aminosyre-sekvenser, en bogstavkombination eller et udsnit af en tidsserie.

Definitionen på et skjult lag er, at det hverken er et inputlag eller outputlag, det vil sige omgivelserne kan ikke "se" et skjult lag, og laget er ikke i direkte forbindelse med omgivelserne. Det er i de skjulte lag, at størstedelen af beregningerne foregår. Skjulte neuroner har ofte en sigmoid aktiveringsfunktion.

Outputneuroner er de neuroner, der overfor omgivelserne angiver resultatet af netværkets behandling af inputtet. Det kan for eksempel være ved at knytte en handling til outputneuronen, når det output, den udsender, er over en bestemt værdi. Handlingen kunne for eksempel være en lampe, der lyser mere eller mindre stærkt, udførslen af en mekanisk funktion eller man kunne tage outputtet direkte som værdien af en tidsserie til et bestemt tidspunkt.

**Signalpropageringsmåden** er nært knyttet netværkets arkitektur, men kan betragtes som et mere globalt princip. Princippet udtaler sig om, hvordan inputsignalet bevæger sig frem mod outputsignalet. Man kan inddele propageringsmåderne i to grupper: en hvor signalet kan vende tilbage til en neuron, det tidligere er udsendt fra, dette kaldes *feed-back* måden eller den *rekursive* måde og en hvor signalet bevæger sig direkte fra inputneuronen til outputneuronen, dette er *feed-forward* måden. Det er den sidste måde, der er den mest anvendte, og det er også den, som vil blive brugt i denne rapport. Feed-forward netværk, der er lagdelte kaldes også for *perceptroner*.

**Oplæringsreglen.** For at et neuralt netværk skal få en bestemt funktion, er det nødvendigt, at det bliver oplært i denne funktion. Der findes mange måder at gøre dette på, se for eksempel Lippmann, 1987. Den, der anvendes i denne rapport, er den såkaldte *back-propagation*- eller *delta-regel*. Det, der back-propageres, er fejlen, netværket begår, når det bliver udsat for et bestemt input/output-par (herom meget mere i næste afsnit).

For at resumere har et neuralt netværk følgende karakteristika:

- Neuronen:
  - *Typen* af de enkelte neuroner
  - *Aktiveringsfunktionen* for de enkelte neuroner
- Netværket:
  - *Arkitekturen*
  - *Systemets tilstand*



- *Input*
- *Output*
- *Oplæringsreglen*

### 3 Delta-reglen

I dette afsnit vil *delta-reglen* blive gennemgået. Delta-reglen er udledt mange steder, se for eksempel Lapedes og Farber (1987), Rumelhart *et al.* (1986) og Lautrup (1989). Delta-reglen kaldes også for *Widrow-Hoff-reglen* efter de to forskere, der udledte den første simple version, uden skjulte lag (Widrow and Hoff, 1960). Den mest almindelige betegnelse er dog *back-propagation* eller blandt "indviede" *back-prop*. Reglen benyttes til at oplære netværket således, at den fejl, det begår, minimeres. Metoden, reglen bygger på, er "steepest descent". I tre dimensioner kan den illustreres ved, at en skiløber skal prøve at finde ned i en dal fra toppen af et bjerg i tæt tåge. Det, denne skiløber gør, hvis han benytter sig af steepest descent metoden, er at han hele tiden går et skridt i den retning, hvor der er stejlest<sup>6</sup>. Matematisk kan "steepest descent" udtrykkes på følgende måde: En funktion  $S(x_1, x_2, \dots, x_n)$  minimeres ved for alle  $x_i$  at udføre følgende iteration

$$x_i^{(t+1)} = x_i^{(t)} - \epsilon \frac{\partial S}{\partial x_i}$$

hvor  $\epsilon$  er lille og positiv<sup>7</sup>. Iterationen startes med gætte  $x_i^{(0)}$ .

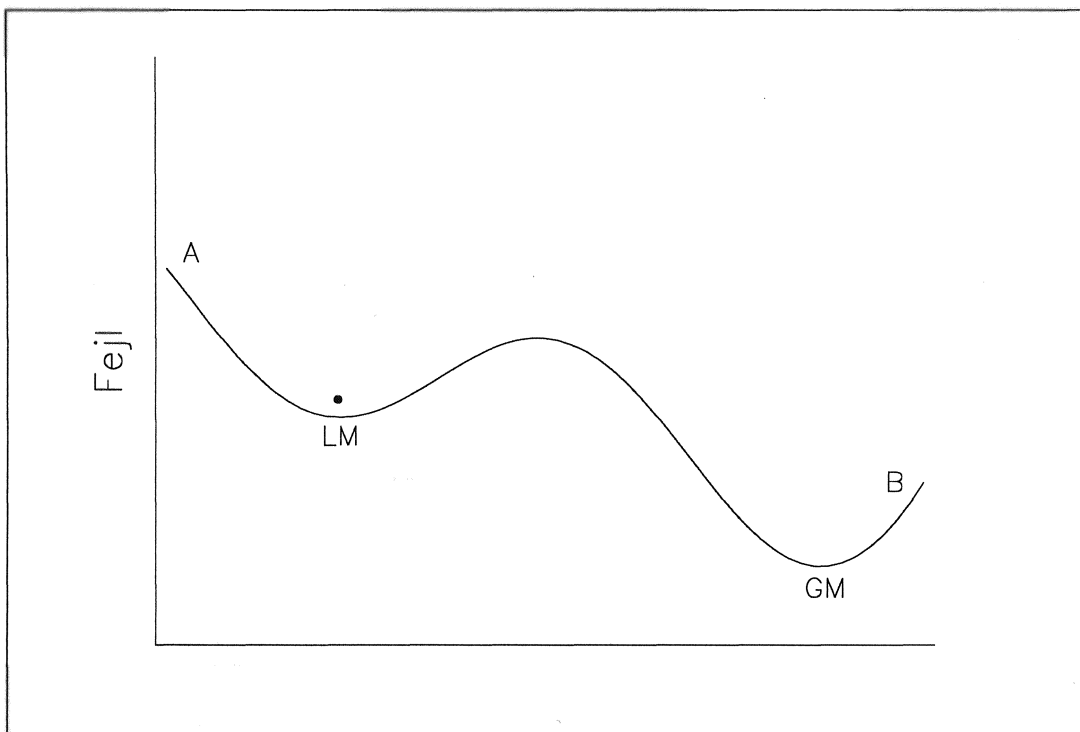
Problemet med denne metode er, at den finder det *nærmeste* minimum og ikke det *globale*. Har funktionen kun ét minimum, er dette ikke noget problem, men har den flere, risikerer man, at det er et *lokalt* minimum, der findes. Startes søgningen i A, se Figur 4, havner den i et lokalt minimum. Startes metoden derimod i B, findes det globale.

Ifølge Rumelhart *et al.* er det sjældent, at man havner i et lokalt minimum, se Rumelhart *et al.* (1986) kapitel 8, det er også den erfaring dette projekt har givet. Eksisterer de alligevel, kan man (måske) slippe udenom dem ved at starte med andre begyndelsesværdier.

Er det sådan, at man ved, at problemet er "svært", i den forstand at der eksisterer lokale minima, kan man anvende en metode udviklet af Kirkpatrick *et al.* (1983). Metoden bygger på det *fysiske* faktum, at ønsker man at finde en tilstand af et metal med meget lav energi, så er den bedste strategi at smelte det og siden langsomt reducere dets temperatur, dette kaldes "annealing" eller på dansk udglødning. Metoden kaldes derfor *simulated annealing*. En intuitiv forklaring, på hvorfor dette virker, kan fås ved igen at se på Figur 4. Man gør det, at man betragter kuglen (tilstanden i parameterrummet), der på figuren er i punktet LM, som et atom, den har dermed mulighed for at udføre varmesvingninger - jo højere temperatur, jo større svingninger. Er svingningerne tilstrækkeligt store, kan man se, at kuglen kan hoppe fra det lokale minimum (LM) til det globale (GM) - men også fra GM tilbage til LM, dog er der størst sandsynlighed for, at kuglen hopper fra LM til GM, da barrieren (i dette tilfælde) er mindst dér. Nu sænkes temperaturen gradvist (systemet relaxeres) på en sådan måde, at det er mest sandsynligt, at kuglen havner i GM. Den ligning, man benytter til at bestemme overgangssandsynlighederne ud fra, er Boltzmanns ligning, derfor kaldes neurale

<sup>6</sup> Erfarne skiløbere fortæller, at hvis man gør som hér beskrevet, så er der stor sandsynlighed for, at man brækker benet, men ideen er god nok !

<sup>7</sup> Skal det være helt korrekt, skal  $\epsilon$  være infinitesimal, men så tager det uendelig lang tid at finde minimummet.



Figur 4. Illustration af et af problemerne ved steepest descent. Grafen er et simpelt billede af error-surface'en, det vil sige fejlen som funktion af parametrene, se tekst for detaljer.

netværk, der benytter sig af dette princip, for *Boltzmann maskiner*. I det virkelige liv er det (desværre) ikke sådan, at fejlen kun afhænger af én parameter, ofte afhænger den af størrelsesordenen 100 eller flere. Dette gør, at barriererne degenererer, sådan at der er flere muligheder for at slippe væk fra et minimum. Det viser sig dog, at jo mere degenereret barriererne er, jo bedre virker simulated annealing.

Nu udledes delta-reglen, bemærk, at udledningen er så generel, at den gælder for *alle* aktiveringsfunktioner, specielt funktionerne  $f(x) = 0.5(1 + \tanh x)$  og  $f(x) = x$ . Den størrelse, der skal minimeres, er den totale fejl,  $E_{tot}$ , defineret som summen af kvadratet på differencen mellem outputtet fra netværket og den forventede værdi:

$$E_{tot} = \sum_p E^p$$

hvor

$$E^p = \sum_i (O_i^p - targ_i^p)^2 \quad (1)$$

$E^p$  er et mål for fejlen begået ved præsenteringen af det  $p$ 'te input/output-par,  $O_i^p$  er det *faktiske* output fra den  $i$ 'te outputneuron og  $targ_i^p$  er det *ønskede* output fra den  $i$ 'te outputneuron.

Vi skal nu finde udtrykket for den afledte af  $E_{tot}$  med hensyn til alle vægtene,  $V_{kj}$ , og tærsklerne,  $T_k$ , af de enkelte neuroner i netværket, da  $E_{tot}$  er en funktion af begge disse<sup>8</sup>. Subscript  $k$  indikerer den neuron der fokuseres på og subscript  $j$

<sup>8</sup>Man kan formelt slå vægte og tærskler sammen. Dette gøres ved at forbinde hver neuron med en neuron, der udsender en konstant værdi. Vægten fra den konstante neuron multipliceret med dens output er så den modtagende neurons tærskel. Dette er ikke gjort her, fordi der er en begrebsmæssig forskel på vægte og tærskler.

løber over alle de neuroner, der giver input til denne. Af definitionen på  $E_{tot}$  ses det, at

$$\frac{\partial E_{tot}}{\partial x} = \sum_p \frac{\partial E^p}{\partial x}$$

hvor  $x$  er enten  $V_{kj}$  eller  $T_k$ . For at beregne de afledte af  $E^p$  anvendes en hjælpe størrelse,  $net_k^p$ , der er givet ved

$$net_k^p = \sum_j V_{kj} O_j^p - T_k$$

hvor  $O_j = f_j(net_j^p)$  er outputtet fra den  $j$ 'te neuron, og  $f_j$  dens aktiveringsfunktion.  $E^p$  kædedifferentieres nu med hensyn til  $net_k^p$  og  $x$ :

$$\frac{\partial E^p}{\partial x} = \frac{\partial E^p}{\partial net_k^p} \frac{\partial net_k^p}{\partial x} \quad (2)$$

Den afledte af fejlen med hensyn til  $x$  bliver altså splittet op i to: én, der har med ændringen af fejlen som funktion af ændringen af nettoinput'et at gøre, og én, der har med ændringen af nettoinput'et som funktion af  $x$  (det vil sige enten  $V_{kj}$  eller  $T_k$ ) at gøre. Anvendes definitionen af  $net_k^p$ , giver den sidste afledte, hvor  $x$  erstattes med henholdsvis  $V_{kj}$  og  $T_k$ :

$$\frac{\partial net_k^p}{\partial V_{kj}} = \frac{\partial}{\partial V_{kj}} \left( \sum_l V_{kl} O_l^p - T_k \right) = O_j^p$$

og

$$\frac{\partial net_k^p}{\partial T_k} = \frac{\partial}{\partial T_k} \left( \sum_l V_{kl} O_l^p - T_k \right) = -1.$$

Nu defineres

$$\delta_k^p = -\frac{\partial E^p}{\partial net_k^p}.$$

Ligning (2) kan da skrives (igen med  $x$  erstattet):

$$-\frac{\partial E^p}{\partial V_{kj}} = \delta_k^p O_j^p.$$

og

$$-\frac{\partial E^p}{\partial T_k} = -\delta_k^p.$$

Nu skal  $\delta_k^p$  beregnes. Det er kun nødvendigt at beregne  $\delta_k^p$  for outputneuroner og skjulte neuroner, da inputneuroner per definition udelukkende virker som modtagere af input fra omgivelserne og altså ikke påvirker input'et direkte. Inputneuroner har derfor ingen (eller med de faste værdier henholdsvis 1 og 0) vægte og tærskler.

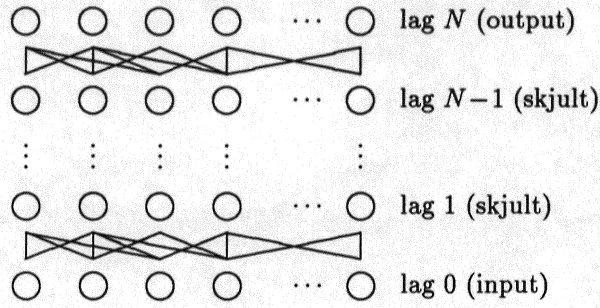
Vi anvender følgende generaliserede netværk, der består af  $N + 1$  lag neuroner, inputlaget har nummer 0 og outputlaget nummer  $N$ , se Figur 5.

$M_n$  defineres som mængden af numre på neuroner i det  $n$ 'te lag. I det følgende er index  $p$  undladt for overskuelighedens skyld.

Antages det, at den  $k$ 'te neuron ligger i lag  $n$ , det vil sige  $k \in M_n$  og anvendes definitionen af  $E$  fås

$$\begin{aligned} \delta_k &= -\frac{\partial}{\partial net_k} \left( \sum_{i \in M_N} (O_i - targ_i)^2 \right) \\ &= -\sum_{i \in M_N} \frac{\partial}{\partial net_k} ((O_i - targ_i)^2) \end{aligned}$$

Udføres differentiationen med hensyn til  $net_k$  kan vi skrive



Figur 5. Neurtalt netværk, bestående af  $N + 1$  lag neuroner. Lag 0 er inputlaget og lag  $N$  er outputlaget. Det er kun nogle få af forbindelserne, der er tegnet, ellers ville det blive alt for uoverskueligt.

$$\delta_k = - \sum_{i \in M_N} 2(O_i - targ_i) \frac{\partial}{\partial net_k} (f_i(net_i) - targ_i),$$

hvor definitionen på  $O_i$  er anvendt. Der differentieres nu med hensyn til  $net_k$  igen:

$$\delta_k = - \sum_{i \in M_N} 2(O_i - targ_i) f'_i(net_i) \frac{\partial net_i}{\partial net_k}.$$

Hvis  $n = N$ , det vil sige hvis den  $k$ 'te neuron ligger i outputlaget, fås:

$$\delta_i = -2(O_i - targ_i) f'_i(net_i), \text{ for } k \in M_N$$

Bemærk, at outputneuroner ofte er lineære, så  $f'_i = 1$ , for alle  $i$ . Er derimod  $n \neq N$ , fås:

$$\delta_k = \sum_{i \in M_N} \delta_i \frac{\partial net_i}{\partial net_k},$$

idet det nyligt fundne udtryk for  $\delta_i$  for en neuron i outputlaget er anvendt. Anvendes definitionen af  $net_i$  fås:

$$\begin{aligned} \delta_k &= \sum_{i \in M_N} \delta_i \frac{\partial}{\partial net_k} \left( \sum_{m \in M_{N-1}} V_{im} O_m - T_i \right) \\ &= \sum_{i \in M_N} \delta_i \sum_{m \in M_{N-1}} \frac{\partial}{\partial net_k} (V_{im} f_m(net_m)) \end{aligned}$$

hvor definitionen af  $O_m$  igen er anvendt. Nu differentieres med hensyn til  $net_k$ :

$$\delta_k = \sum_{i \in M_N} \delta_i \sum_{m \in M_{N-1}} V_{im} f'_m(net_m) \frac{\partial net_m}{\partial net_k}. \quad (3)$$

Hvis den  $k$ 'te neuron ligger i lag  $N - 1$  ( $n = N - 1$ ), fås:

$$\delta_k = \sum_{i \in M_N} \delta_i V_{ik} f'_k(net_k) = f'_k(net_k) \sum_{i \in M_N} \delta_i V_{ik} \text{ for } k \in M_{N-1}$$

Dette ses at kunne generaliseres til følgende formel for  $\delta_k$  i det  $n$ 'te lag, hvor laget er et skjult lag, er:

$$\delta_k = f'_k(net_k) \sum_{i \in M_{l+1}} \delta_i V_{ik} \text{ for } k \in M_l, 1 \leq l \leq N - 1 \quad (4)$$



Bemærk, ofte er  $f_k = f$ , for  $k \notin M_N$ . Man bruger altså den samme aktiveringsfunktion for alle skjulte neuroner.

Af Ligning (4) ses, at man ved beregning af  $\delta$  'erne skal starte i outputlaget og så arbejde sig bagud mod inputlaget. Det vil sige at fejlen sendes modsat inputtet, eller den *back-propageres*.

For at minimere  $E_{tot}$  kan det nu ses at følgende iterative metode skal anvendes på  $V_{kj}$  og  $T_k$ :

$$V_{kj}^{(ny)} = V_{kj}^{(gammel)} + \Delta V_{kj}$$

og

$$T_k^{(ny)} = T_k^{(gammel)} + \Delta T_k$$

hvor

$$\Delta V_{kj} = \epsilon_V \left( -\frac{\partial E_{tot}}{\partial V_{kj}} \right) = \epsilon_V \sum_p \delta_k^p O_j^p$$

og

$$\Delta T_k = \epsilon_T \left( -\frac{\partial E_{tot}}{\partial T_k} \right) = -\epsilon_T \sum_p \delta_k^p$$

hvor

$$\delta_k^p = -2(O_k^p - targ_k^p) f'_k(net_k^p),$$

hvis den  $k$ 'te neuron er i outputlaget og

$$\delta_k^p = f'_k(net_k^p) \sum_{i \in M_{I+1}} \delta_i^p V_{ik},$$

hvis den  $k$ 'te neuron ikke er i outputlaget. Vægtene og tærsklerne opdateres altså efter at *alle* input'ene er blevet præsenteret.

Bemærk, at delta-reglen i princippet ikke kræver nogen nedre grænse, hverken for hvor mange input/output-par, der skal være, eller for hvor mange gange de skal præsenteres for nettet, for at den virker. Er der for eksempel kun ét input/output par, er det muligt at beregne ændringerne af alle vægte/tærskler, og dermed er det muligt at optræne nettet. Præsenteres dette par kun én gang bliver fejlen nettet begår selvfølgelig stor. Antallet af input/output-par siger noget om, hvor generelt man ønsker at præsentere tidsserien for nettet, og antallet af gange input/output-parrene præsenteres, siger noget om, hvor nøjagtigt nettet skal lære tidsserien. De to størrelser afhænger delvist af hinanden, for er problemet svært (det vil sige komplekst), er det nødvendigt både at have mange input/output-par for at beskrive problemet udtømmende, men også at præsentere dem mange gange for nettet for at nøjagtigheden bliver stor. Ønskes det, at nettet skal generalisere, er det nødvendigt, at antallet af præsentationer i det mindste er større end antallet af ubekendte,  $U$ , det vil sige antallet af vægte og tærskler, givet ved

$$U = \sum_{n=1}^N \#M_n + \sum_{n=1}^N (\#M_n \#M_{n-1})$$

hvor  $\#M_n$  betegner antallet af elementer i mængden  $M_n$ . Den første sum er antallet af tærskler og den anden er antallet af vægte.

Problemet med "steepest descent" er, at algoritmen ikke er særlig effektiv. Skal den for eksempel finde ned i bunden af en lang smal "dal", så går den i siksak, istedet for at gå direkte, da gradienten altid er vinkelret på den foregående. Der findes da også andre metoder til at minimere funktioner, en af dem er den såkaldte "conjugate gradient descent", der minder om steepest descent, men hvor ændringen af vægte og tærskler bestemmes af, at man ønsker at reducere *fejlen* med en vis procentdel af dens nuværende værdi. Måden dette fremkommer på er, at den retning, man går i, blot bliver *bøjet* i retning af gradienten. De to vigtigste algoritmer

er *Fletcher-Reeves* og *Polak-Ribiere* algoritmerne. For en nærmere gennemgang af "conjugate gradient descent" se for eksempel Makram-Ebeid et al. (1989), hvor der også bliver beskrevet en videreudvikling af metoden. Denne videreudvikling kaldes "bounded conjugate gradient descent", og ideen er at begrænse ændringen af amplituderne - der kan blive meget store ved "conjugate gradient descent" - for at undgå numerisk instabilitet. En anden minimeringsalgoritme er "stochastic gradient descent", hvor tærskler og vægte ændres for hver præsentation, og ændringen er proportional med fejlen,  $E_t = (y_t^* - y_t)^2$ , således, at er fejlen stor, vil ændringen af tærskler og vægte også være stor. Samtidig ændres  $\epsilon$  også for hver præsentation. Endvidere bliver der også arbejdet på at optimere back-propagation ved forskellige trick. En ide er at lade  $\epsilon$  variere med hvor god den sidste korrektion af fejlen har været, bliver fejlen reduceret meget gøres  $\epsilon$  større og reduceres den lidt eller stiger den, da sættes  $\epsilon$  ned.

En helt anden synsvinkel som Owens og Filkin (Owens and Filkin, 1989) har brugt, er at betragte vægte og tærskler som kontinuerte funktioner af tiden. Minimerings problemet bliver da til, at man skal løse nogle koblede differential-ligninger, der ofte er matematisk stive. Indenfor de sidste år er der kommet nogle meget kraftfulde løsningspakker til netop dette problem, anvendes disse, findes der hurtigt (eller i hvert tilfælde hurtigere end steepest descent for nogle problemer) et mindste kvadraters minimum. Bagdelen ved denne metode er, at den kræver meget stor lagerplads, fordi diskretiseringsmetoden er "backward Euler", der er en *implicit* metode, som dermed kræver, at alle de resulterende differensligninger gemmes og løses på én gang til sidst.

Det er interessant, at man ikke indenfor neurobiologien har fundet en oplæringsmekanisme på det mikroskopiske (neuron) niveau, der ligner back-propagation, se for eksempel Rumelhart et al. (1986) kapitel 4. Eftersøgningen pågår dog stadig, og Stork and Hall (1989) har konstrueret en model for hvordan et biologisk netværk kunne se ud, hvis det skulle være i stand til at foretage back-propagation. Der er dog ingen tvivl om, at man - på det makroskopiske plan - altid lærer af sine fejl!

## 4 Indledende eksperimenter med neurale netværk

I de indledende eksperimenter med de neurale netværk vil de blive oplært i nogle simple tidsserier. Formålet med dette er at se, om netværkene i det hele taget kan lære tidsserier. Dette må betragtes som værende en forudsætning for, at de kan lære og efterfølgende forudsige forløbet af den mere komplicerede temperaturserie.

### 4.1 Generelt om eksperimenterne

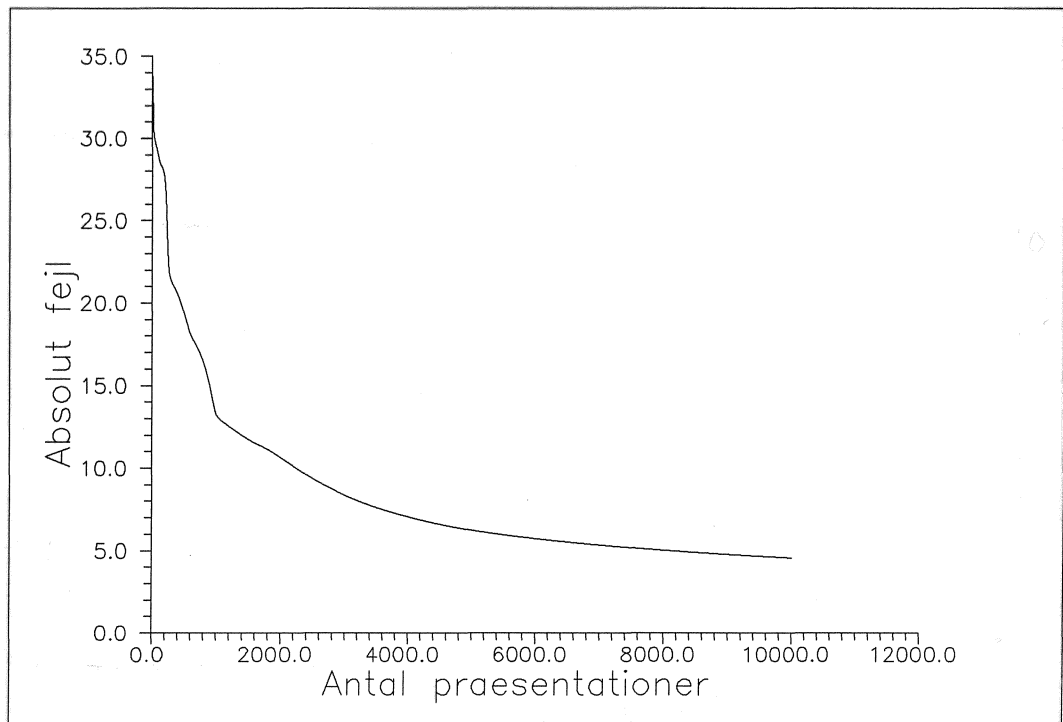
Måden, et netværk skal forudsige en tidsserie på, vil helt generelt være, at man præsenterer det for et udsnit af den gamle (kendte) tidsserie:  $z_{t-N\Delta}, \dots, z_{t-2\Delta}, z_{t-\Delta}, z_t$ , hvor  $z_t$  betegner værdien af tidsserien til tidspunktet  $t$  og  $\Delta$  er tidsenheden, det skal så forudsige værdien af tidsserien til tid  $t + \tau$ ,  $\tau = T\Delta$ . Det skal ikke bruges til at forudsige tidsserien til flere tidspunkter ud i fremtiden, da dette vil "forvirre" netværket. Ønsker man at forudsige til flere forskellige tider, må man derfor oplære flere netværk. Alternativt kunne netværkets output bruges som input til beregningen af værdien af tidsserien i det næste tidsskridt og så fremdeles, indtil man når det ønskede tidspunkt. Bagdelen ved dette er, at den fejl, der uundgåeligt er ved forudsigelsen, vil blive ophobet. I sidste del vil begge

metoder dog blive forsøgt.

Det er altså ikke selve tidsserien  $(t; z_t)^9$ , men derimod dens *Poincaré map*  $(z_{t-N\Delta}, \dots, z_{t-2\Delta}, z_{t-\Delta}, z_t; z_{t+\tau})$ , netværket skal oplæres i. Mappen er med andre ord, det rum, der udspændes af  $z_{t-N\Delta}, \dots, z_{t-2\Delta}, z_{t-\Delta}, z_t$  og  $z_{t+\tau}$ . Der er altså tale om en afbildning fra  $R^{N+1}$  ind i  $R$ . I kaosteorien kaldes dannelsen af map'en for "imbedding". Det, at netværket lærer map'en, gør, at man kan bruge neurale netværk til at give en analytisk approksimation til funktionens map, for eksempel indenfor kaos-fysikken. Det er jo bare at skrive aktiveringsfunktionerne op og indsætte de fundne værdier for vægte og tærskler fra det oplærte netværk.

De enkelte eksperimenter vil alle have følgende generelle udseende: Netværket bliver præsenteret for 1,000 input/output-par<sup>10</sup>. Denne præsentation fortsætter indtil netværket er blevet præsenteret for input/output-parrene 10,000 gange<sup>11</sup>.

Følger man optræningen, ses det, at fejlen meget hurtigt reduceres, hvorefter netværket nærmer sig asymptotisk til et eller andet minimum - ofte 0, se Figur 6.



Figur 6. Typisk forløb af den totale absolutte fejl som funktion af antallet af præsentationer, den absolutte fejl er givet ved  $E_{tot} = \sum_{p=1}^{1000} |O^p - targ^p|$ , hvor  $O^p$  er det faktiske output, og  $targ^p$  er det ønskede output af det  $p$ 'te input/output par.

En parameter, der er meget vigtig for hvor hurtigt fejlen konvergerer, er  $\epsilon$  (se afsnittet om delta-reglen). Der findes ingen måde, hvorpå man på forhånd kan bestemme  $\epsilon$ . Derfor er en vigtig bestanddel af optræningen, at finde en passende værdi for  $\epsilon$ . Ved at variere  $\epsilon$  og studere forløbet af den resulterende fejl, får man hurtigt en fornemmelse af størrelsen af  $\epsilon$ . Det er sådan, at  $\epsilon$  for stor, så "eksploderer" fejlen, fordi de skridt, som vægtene og tærsklerne skal ændres med, bliver så store, at delta-reglen ikke kan "holde" fejlen på rette spor. Det kan være lidt svært

<sup>9</sup> Afbildningen  $(t; z_t)$  anvendes, hvis man ønsker at anvende neurale netværk til at interpolere en funktion, se for eksempel Shustrorovich (1989).

<sup>10</sup> Der er ingen speciel grund til at anvende 1,000 input/output-par, det kunne ligeså godt have været for eksempel 100 eller 5,000.

<sup>11</sup> Dette tal er også arbitrært, men som det ses af Figur 6, skal det være et tal af denne størrelsesorden, for at fejlen bliver tilstrækkelig lille.



at forestille sig hvordan fejlen ligefrem kan eksplodere; numerisk instabilitet giver sig ofte udtryk i oscillationer. Er proportionalitetsfaktoren af en vis størrelse, kan dette faktisk ske, for jo længere væk fra minimummet man kommer, jo større bliver gradienten og dermed skridtet, hvilket bringer tilstanden endnu længere væk, og så videre. En måde at dæmpe denne numeriske instabilitet på er at addere en vis brøkdel af den sidste vægt/tærskelændring til den aktuelle ændring. Dette nye led kaldes *momentum* leddet.

Er  $\epsilon$  derimod for lille, er det tydeligt, at fejlen konvergerer meget langsomt, typisk ændrer den sig kun 0.01 % per præsentation eller slet ikke. Det er imellem disse ekstremer, at det ønskede  $\epsilon$  skal findes. Det, at  $\epsilon$ 'erne er fundet ved en vurdering, gør, at selv om alle eksperimenterne køres det samme antal gange, så kan det være svært at sammenligne resultaterne. Det er jo ikke sikkert, at det er den optimale  $\epsilon$ , der er anvendt, hver gang. Dog er netværkene ofte meget følsomme overfor ændringer i  $\epsilon$ . For eksempel eksploderede netværk 1b under optræningen i tidsserie F for  $\epsilon = 7 \cdot 10^{-8}$  og der var meget langsom konvergens for  $\epsilon = 5 \cdot 10^{-8}$ . Som det ses af Tabel 2 gav  $\epsilon = 6 \cdot 10^{-8}$  god konvergens.

## 4.2 Evaluering af resultaterne

Evalueringen af det optrænede netværk bliver foretaget ved, at det bliver præsenteret for 500 nye input/output-par, som det aldrig har set før<sup>12</sup>. Typisk løber  $t$  fra 1001 til 1500; hvor  $t$  under optræningen løb fra 1 til 1000. For tidsserie D (se Figur 15) blev  $z_0$  dog istedet ændret fra 0.1 til 0.3.

For at kunne vurdere hvor godt et netværk har lært en given tidsserie, opstilles nogle størrelser, der hver på deres måde fortæller noget om fejlen. Bemærk, at alle  $z_t^{(net)}$  er dannet ved, at netværket er blevet præsenteret for  $z_{t-\tau}$ . Det er altså kendte (dvs. fra før  $t - \tau$ ) værdier af tidsserien, der anvendes. Størrelserne er:

1. Middelværdien af fejlen, defineret som:

$$m = \frac{1}{N} \sum_{t=1}^N f_t,$$

hvor  $N$  er antallet af forudsigelser og  $f_t = z_t^{(net)} - z_t$ , er fejlen defineret som forskellen mellem den af netværket forudsagte,  $z_t^{(net)}$ , og den faktiske,  $z_t$ , værdi af tidsserien til tiden  $t$ .

2. Den relative fejl<sup>13</sup>,

$$r = \frac{1}{N} \sum_{t=1}^N \frac{f_t}{z_t}.$$

3. Spredningen på fejlen, givet ved:

$$s = \sqrt{\frac{\sum_{t=1}^N (f_t - m)^2}{N - 1}}.$$

4. Korrelationskoefficienten,

$$k = \frac{\sum_{t=1}^N z_t z_t^{(net)}}{\sqrt{\sum_{t=1}^N z_t^2 \sum_{t=1}^N z_t^{(net)2}}}.$$

<sup>12</sup>Dette er ikke altid muligt. For eksempel for periodiske funktioner, hvor netværket er blevet optrænet i hele perioden.

<sup>13</sup>Der er visse problemer ved denne størrelse. Er for eksempel værdien af  $z_t$  tæt ved eller lig 0, bliver  $r$  meget stor bare  $z_t^{(net)}$  er lidt forskellig fra nul. Er  $z_t = 0$  ses der helt bort fra dette bidrag.

5. Index'et, indført af Lapedes og Farber (1987), defineret ved:

$$i = \frac{s}{\sigma},$$

hvor  $\sigma$  er spredningen på den *faktiske* tidsserie, givet ved:

$$\sigma = \sqrt{\frac{\sum_{t=1}^N (z_t - \mu)^2}{N - 1}}$$

og

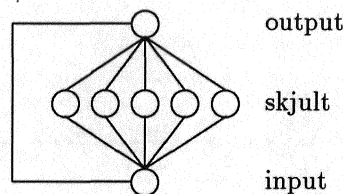
$$\mu = \frac{1}{N} \sum_{t=1}^N z_t$$

Da  $\sigma$  siger noget om hvor stor afvigelsen af funktionen fra dens middelværdi er, eller noget om hvor "vildt"  $z_t$  opfører sig, og  $s$  siger noget om overensstemmelsen mellem  $z^{(net)}$  og  $z$ , er  $i$  lille, når  $s$  er lille og  $\sigma$  er stor, det vil sige når  $z$  fluktuerer meget, og det alligevel er lykkedes  $z^{(net)}$  at følge den.

### 4.3 Eksperimenterne

I de indledende eksperimenter optrænes 4 forskellige netværk:

1. Netværk 1a (se Figur 7), bestående af 1 inputneuron, 1 skjult lag med 5 sigmoide neuroner og 1 lineær outputneuron. Der er forbindelse mellem inputneuronen og det skjulte lag og mellem det skjulte lag og outputneuronen.
2. Netværk 1b (se Figur 7), som netværk 1a, der er dog også forbindelse direkte fra inputneuronen til outputneuronen.

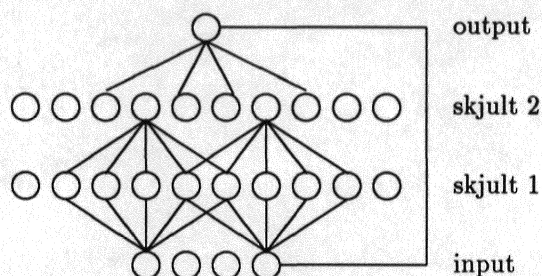


Figur 7. Netværk 1a+b, bestående af 1 lineær inputneuron, 5 skjulte sigmoide neuroner og 1 lineær outputneuron. Forskellen på netværk 1a og 1b er, at b har en direkte lineær forbindelse fra input- til outputneuronen, angivet ved den stiplede linie, hvilket a ikke har.

3. Netværk 2a (Figur 8), bestående af 4 inputneuroner, 2 skjulte lag med hver 10 sigmoide neuroner og 1 lineær outputneuron, *uden* den direkte forbindelse mellem input- og outputlaget.
4. Netværk 2b (Figur 8), som 2a med *med* den direkte forbindelse.

Netværkene er de samme, som Lapedes og Farber (1987) bruger. De forsøges optrænet i 6 forskellige tidsserier: tidsserie A til F.

Tidsserierne er valgt ud fra, at endemålet er, at netværket skal optrænes i en temperaturserie. Når der er tale om en temperaturserie, vil man først og fremmest forvente, at den består af en årlig svingning på grund af Jordens rotationsakses hældning i forhold til Jordens bane om Solen. Det er derfor nødvendigt, at vi undersøger om et neuralt netværk kan modellere *periodiske svingninger*, for eksempel i form af en sinuskurve. Dernæst vil man forvente *trends* i serien, det vil sige en



Figur 8. Netværk 2a+b, bestående af 4 lineære inputneuroner, 2 skjulte lag med hver 10 sigmoide neuroner og 1 lineær outputneuron. Forskellen på netværk 2a og 2b er, at b har en direkte lineær forbindelse fra input- til outputneuronen, angivet ved den stiplede linie, dette har a ikke. Igen er ikke alle forbindelserne tegnet.

jævn(t) stigning/fald over en længere periode, for eksempel i form af en ret linie. Denne trend kunne være et udtryk for drivhuseffektens virkning på temperaturen, hvis den altså giver sig udslag i en temperaturstigning.

For at se, hvordan et neuralt netværk oplæres, starter vi med at prøve at oplære netværk 1a i at forudsige *tidsserie A*, givet ved:

$$z_t = 4.$$

Tærsklerne og vægtene fra det oplærte netværk ses i Tabel 1.

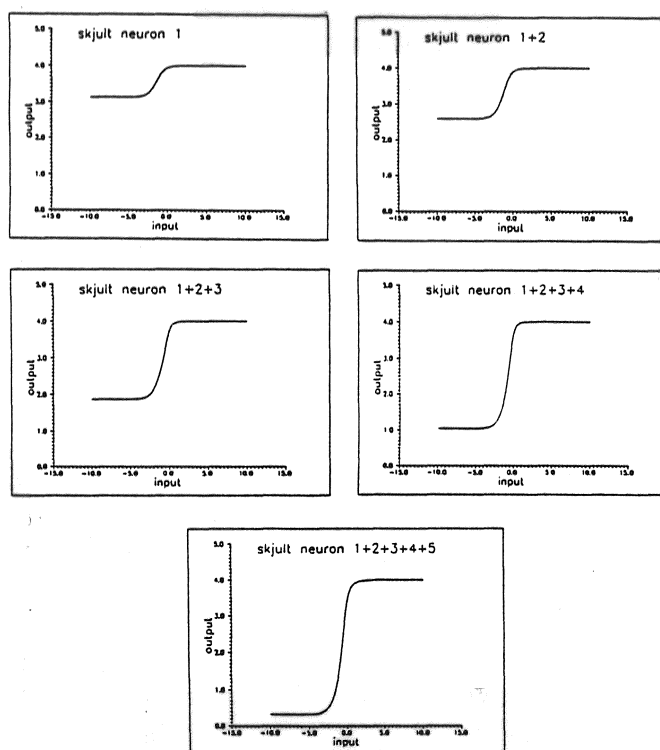
Tabel 1. Netværk 1a og 1b's vægte og tærskler efter optræning i *tidsserie A*. Felterne mærket '-' markerer, at de tilhørende vægte og tærskler ikke anvendes i nettet.

i	Netværk 1a			Netværk 1b		
	$T_i$	$V_{6,i}$	$V_{i,0}$	$T_i$	$V_{6,i}$	$V_{i,0}$
1	1.24	-0.87	-0.87	1.23	-0.87	-1.23
2	1.19	-0.53	-1.23	1.89	-0.53	-1.19
3	0.75	-0.73	-1.89	1.85	-0.73	-0.75
4	1.03	-0.84	-1.85	0.54	-0.84	-1.03
5	0.48	-0.71	-0.69	0.75	-0.73	-0.47
6	-4.00	-	-	0.30	-	1.08

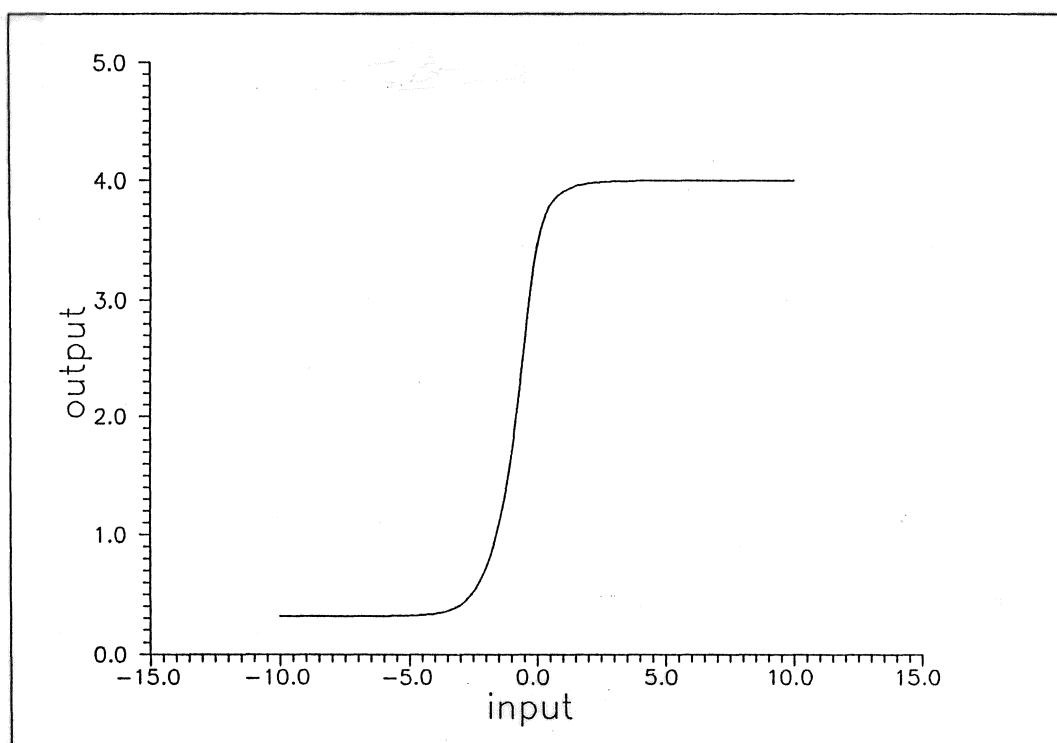
Det er svært at overskue disse værdier, og det bliver sværere jo mere kompliceret netværkets arkitektur bliver. Derfor er den resulterende responsfunktion af de enkelte skjulte neuroner vist i Figur 9 og netværkets totale responsfunktion i Figur 10. Ud fra dette kan man se, at for værdier større end 1 vil nettet svare med værdien cirka 4. Som det ses af Tabel 2, er fejlen mindre end computerens nøjagtighed.

Map'en for  $z_t = 4$  er et punkt (4;4). Sammenligner man dette med den map (=responsfunktion), nettet giver (Figur 10), ses at den også indeholder dette punkt, men at ligegyldigt hvilken værdi, man præsenterer netværket for, giver det et svar. Er værdien omkring 4, bliver svaret også 4.

Kigger vi på, hvordan netværk 1b klarede tidsserien, ser vi, at også dette netværk lærte det ønskede meget nøjagtigt, se Tabel 2. Responsfunktionen viser imidlertid, at netværket lærer på en helt anden måde. I stedet for, som netværk 1 at sige, at de fleste input (i hvert tilfælde dem, der er i nærheden af og større en 4) skal give værdien 4 som output, siger netværk 1b, at det skal være inputværdien, der skal gives som output. Dette ses af Tabel 1, hvor den eneste vægt, der er



Figur 9. Aktiveringsfunktionen for de enkelte skjulte neuroner i netværk 1a oplært i tidsserie A.



Figur 10. Den totale responsfunktion for netværk 1a oplært i tidsserie A.

positiv (fremmende) er vægten mellem input- og outputneuronen,  $V_{6,0}$ . Resten af vægtene er negative (hæmmende).



Tabel 2. Resultater fra optræningen af netværk 1a til 2b i tidsserierne A-F. Definitionen af  $m$  (middelværdien af fejlen),  $r$  (middelværdien af den relative fejl),  $s$  (spredningen på fejlen),  $k$  (korrelationskoefficienten) og  $i$  (index) kan ses på side 21f.  $\epsilon$  er den størrelse, der indgår i korrektionen af vægtene og tærsklerne. i.d. betyder at størrelsen ikke er defineret.

		$m$	$r$	$s$	$k$	$i$	$\epsilon$
1a	A	0.00	0.00	0.00	1.00	i.d.	$1 \cdot 10^{-3}$
	B	$2.24 \cdot 10^2$	$5.90 \cdot 10^1$	$4.34 \cdot 10^1$	$9.93 \cdot 10^{-1}$	1.00	$1 \cdot 10^{-4}$
	C	$2.85 \cdot 10^{-3}$	$-3.00 \cdot 10^{-3}$	$7.26 \cdot 10^{-3}$	1.00	$9.40 \cdot 10^{-3}$	$1 \cdot 10^{-3}$
	D	$-6.18 \cdot 10^{-4}$	$-1.12 \cdot 10^3$	$8.94 \cdot 10^{-3}$	1.00	$2.54 \cdot 10^{-2}$	$1 \cdot 10^{-3}$
	E	$4.34 \cdot 10^{-4}$	$2.22 \cdot 10^{-2}$	$7.96 \cdot 10^{-2}$	$9.97 \cdot 10^{-1}$	$7.95 \cdot 10^{-2}$	$1 \cdot 10^{-4}$
	F	$2.18 \cdot 10^2$	$5.78 \cdot 10^{-1}$	$4.33 \cdot 10^1$	$9.93 \cdot 10^{-1}$	1.00	$1 \cdot 10^{-3}$
1b	A	$1.33 \cdot 10^{-15}$	0.00	0.00	1.00	i.d.	$1 \cdot 10^{-4}$
	B	$-8.14 \cdot 10^{-1}$	$-2.14 \cdot 10^{-3}$	$2.01 \cdot 10^{-1}$	1.00	$4.63 \cdot 10^{-3}$	$6 \cdot 10^{-8}$
	C	$4.42 \cdot 10^{-3}$	$-1.99 \cdot 10^{-3}$	$6.28 \cdot 10^{-3}$	1.00	$8.14 \cdot 10^{-3}$	$5 \cdot 10^{-5}$
	D	$-9.95 \cdot 10^{-4}$	$-3.96 \cdot 10^2$	$9.11 \cdot 10^{-3}$	1.00	$2.53 \cdot 10^{-2}$	$5 \cdot 10^{-4}$
	E	$2.59 \cdot 10^{-4}$	$2.61 \cdot 10^{-1}$	$7.89 \cdot 10^{-2}$	$9.97 \cdot 10^{-1}$	$7.87 \cdot 10^{-2}$	$1 \cdot 10^{-4}$
	F	$5.59 \cdot 10^{-1}$	$3.26 \cdot 10^{-2}$	$2.46 \cdot 10^{-1}$	1.00	$5.68 \cdot 10^{-3}$	$6 \cdot 10^{-8}$
2a	A	$4.44 \cdot 10^{-16}$	0.00	0.00	1.00	i.d.	$1 \cdot 10^{-4}$
	B	$-2.23 \cdot 10^2$	$5.91 \cdot 10^{-1}$	$4.64 \cdot 10^1$	$9.92 \cdot 10^{-1}$	1.00	$1 \cdot 10^{-3}$
	C	$4.52 \cdot 10^{-3}$	$5.51 \cdot 10^{-3}$	$3.01 \cdot 10^{-2}$	$9.99 \cdot 10^{-1}$	$3.90 \cdot 10^{-2}$	$1 \cdot 10^{-4}$
	D	$4.91 \cdot 10^{-3}$	0.00	$3.47 \cdot 10^{-1}$	$8.30 \cdot 10^{-1}$	$9.99 \cdot 10^{-1}$	$2 \cdot 10^{-3}$
	E	$-3.40 \cdot 10^{-4}$	$-3.98 \cdot 10^{-2}$	$2.62 \cdot 10^{-2}$	1.00	$2.61 \cdot 10^{-2}$	$1 \cdot 10^{-4}$
	F	$3.75 \cdot 10^2$	1.00	$5.19 \cdot 10^1$	$-9.90 \cdot 10^{-1}$	1.00	$2 \cdot 10^{-3}$
2b	A	$2.66 \cdot 10^{-15}$	0.00	0.00	1.00	i.d.	$1 \cdot 10^{-5}$
	B	-2.32	$-6.11 \cdot 10^{-3}$	$5.76 \cdot 10^{-1}$	1.00	$1.33 \cdot 10^{-2}$	$6 \cdot 10^{-8}$
	C	$2.13 \cdot 10^{-2}$	$-5.21 \cdot 10^{-2}$	$5.00 \cdot 10^{-2}$	$9.98 \cdot 10^{-1}$	$6.47 \cdot 10^{-2}$	$1 \cdot 10^{-5}$
	D	$4.96 \cdot 10^{-2}$	0.00	$6.80 \cdot 10^{-2}$	$9.93 \cdot 10^{-1}$	$1.93 \cdot 10^{-1}$	$5 \cdot 10^{-4}$
	E	$-4.98 \cdot 10^{-4}$	$2.51 \cdot 10^{-2}$	$9.66 \cdot 10^{-2}$	$9.95 \cdot 10^{-1}$	$9.63 \cdot 10^{-2}$	$1 \cdot 10^{-5}$
	F	-2.48	$-6.52 \cdot 10^{-3}$	$6.37 \cdot 10^{-1}$	1.00	$1.47 \cdot 10^{-2}$	$3 \cdot 10^{-8}$

Den næste tidsserie, vi prøver at oplære netværk 1a i, er tidsserie B, der er en ret linie, givet ved:

$$z_t = 0.3t - 2$$

Som det ses af Tabel 2, kan man ikke sige, at det er lykkedes netværket at lære denne serie. Den manglende overensstemmelse skyldes, at map'en er lineær, se Figur 11, men at input'et kun er forbundet til output'et gennem semi-lineære neuroner. Så nettet prøver at lave en semi-lineær approksimation af noget lineært. Derimod har 1b lært tidsserien. Det skyldes selvfølgelig den direkte (lineære) forbindelse.

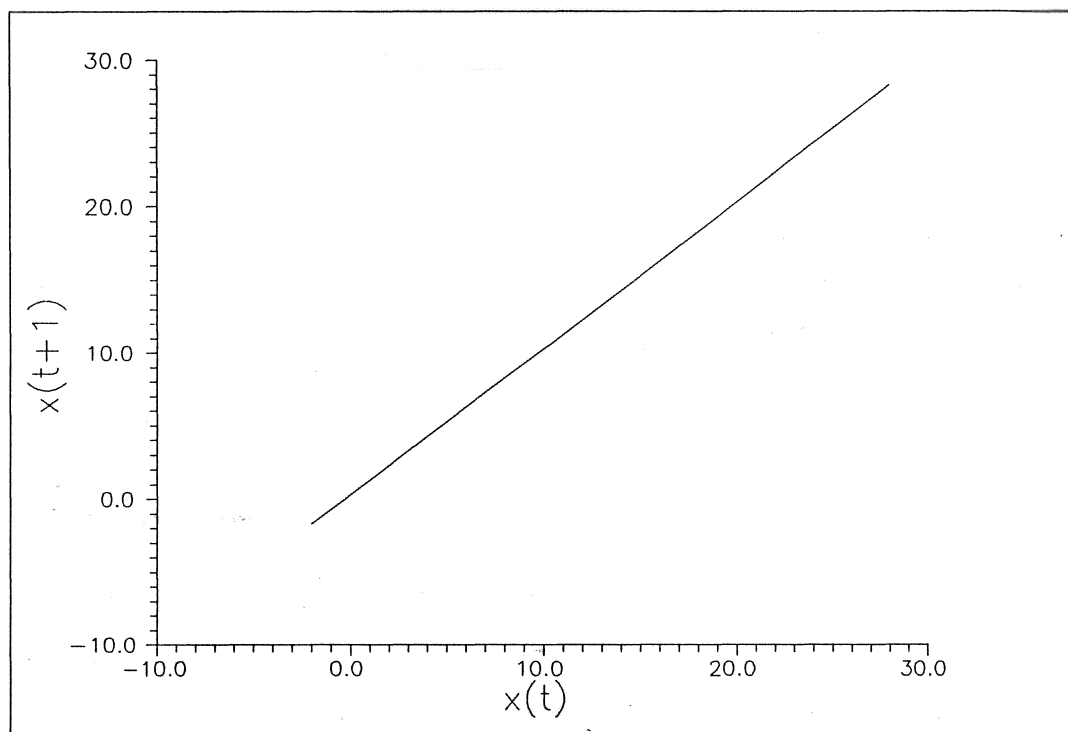
Det vil nu være interessant at se, hvordan netværkene klarer tidsserier, der har en mere kompliceret map. Husk, at det er map'en, netværket skal lære, og ikke funktionen selv. Netværket forsøges derfor optrænet i tidsserie C, der er en sinusfunktion, se Figur 12, givet ved

$$z_t = \sin(0.01t).$$

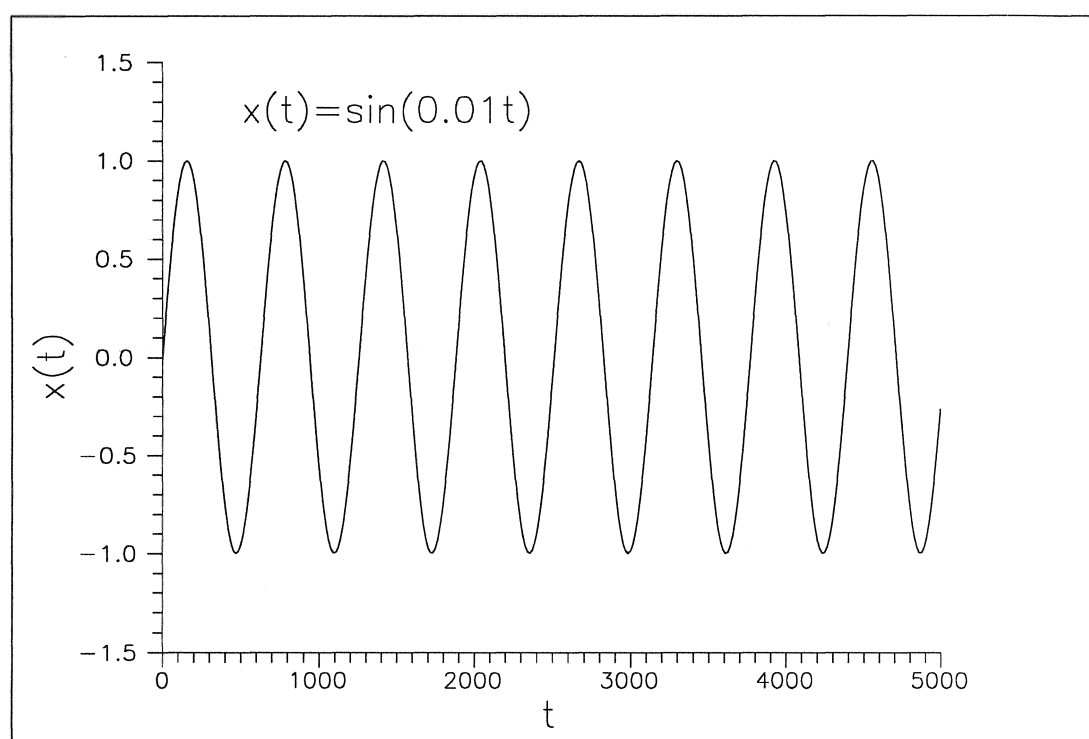
Map'en for denne tidsserie er en meget fladtrykt ellipse med hovedakse sammenfaldende med linien  $y = x$ , se Figur 13.

Ellipsen er faktisk så sammentrykt, at den fejl, man begår ved at approksimere map'en med udsnittet af den førnævnte linie, svarende til det stykke ellipsen dækker, er lille. På grund af at linien er begrænset, er det lykkedes netværk 1a at lære denne tidsserie tilfredsstillende. Som det ses af Figur 14, giver nettet netop



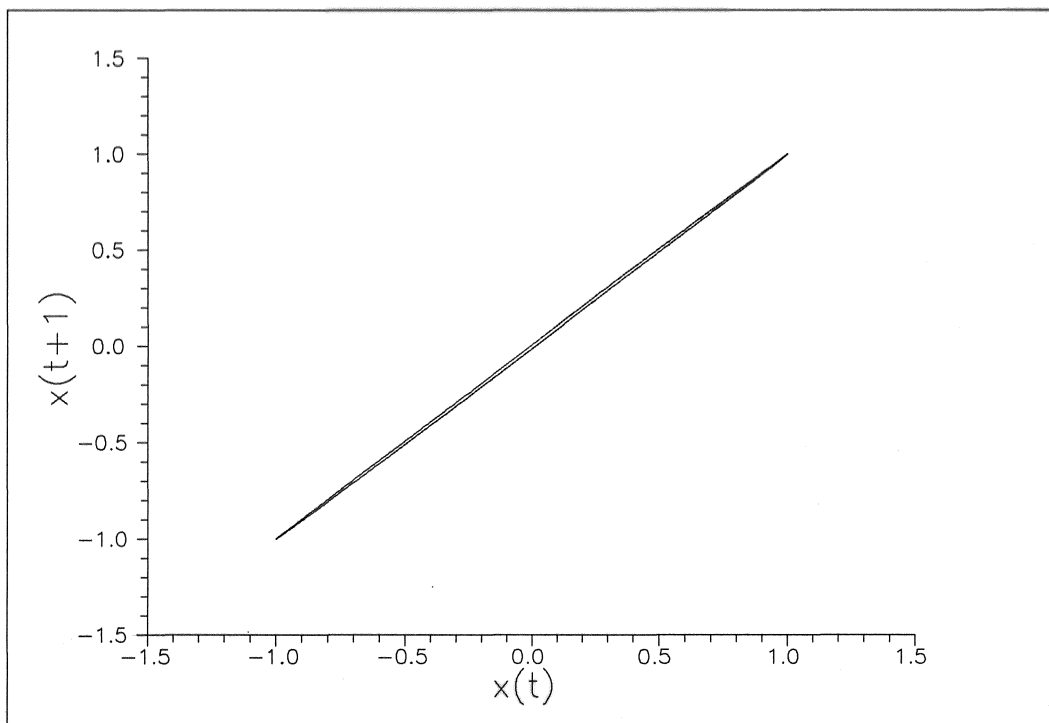


Figur 11. Map'en af tidsserie B.

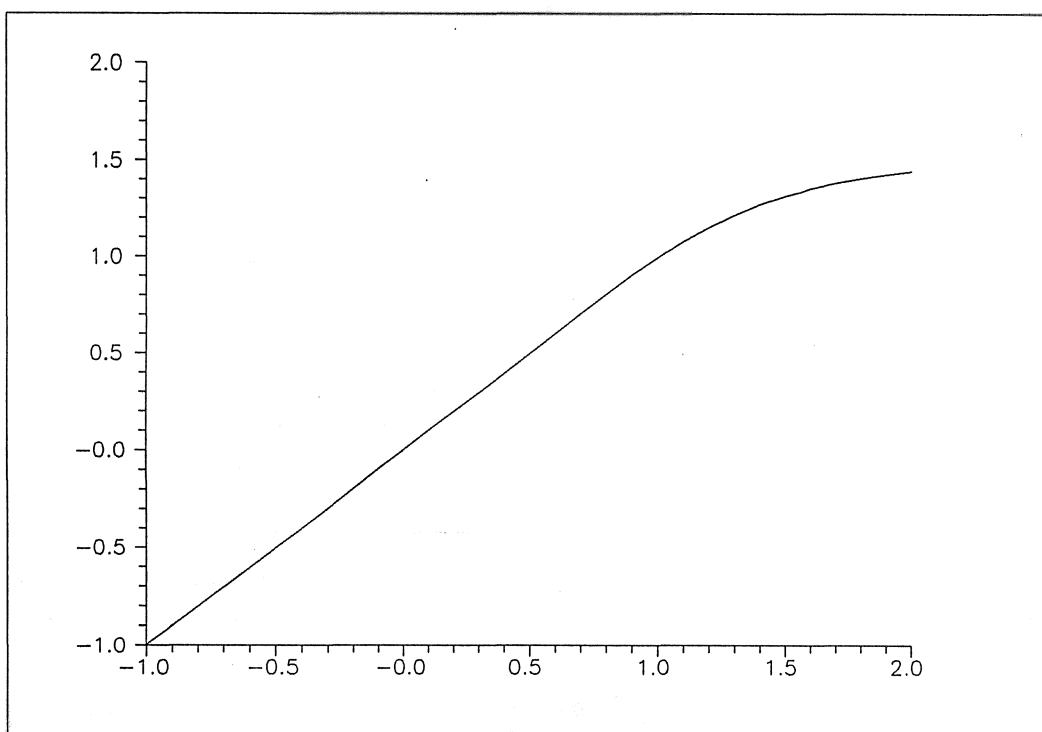


Figur 12. Tidsserie C.

linien  $y = x$  som respons i intervallet fra 0 til 1. Bemærk, at responsfunktionen, betragtet over et større område, har form som de enkelte neuronernes aktiveringsfunktion.



Figur 13. Map'en af tidsserie C.



Figur 14. Netværk 1a's responsfunktion efter oplæring i tidsserie C.

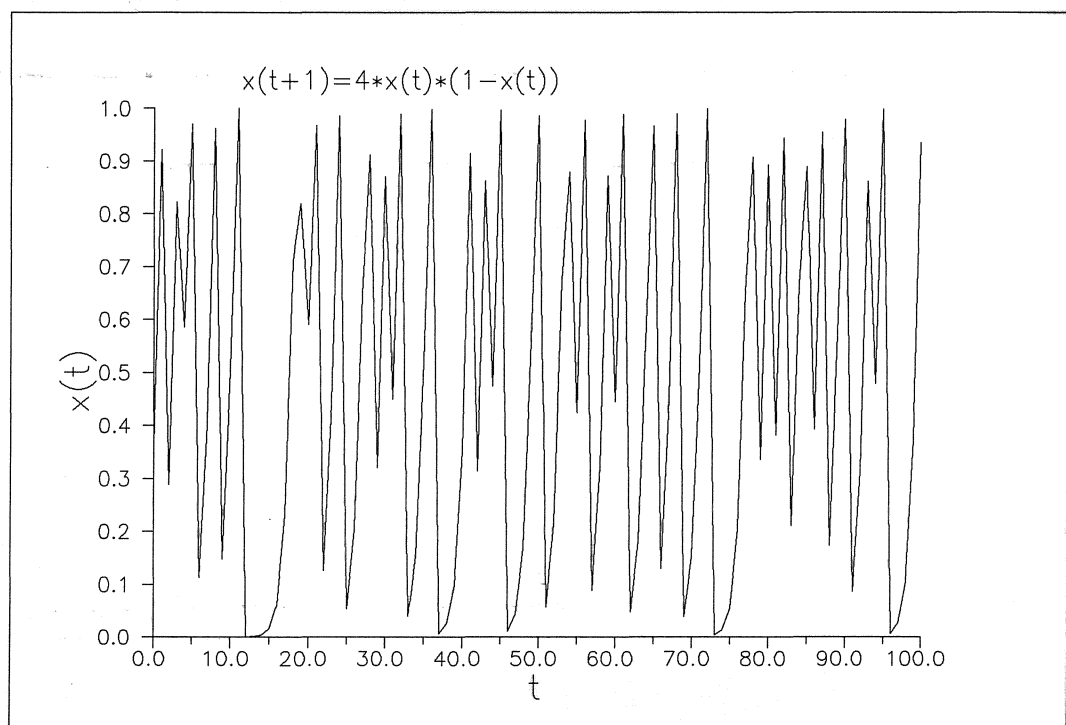
Det gælder helt generelt, at hvis en funktion er kontinuert, og forudsigelsen kun rækker et lille stykke frem i tiden, vil den resulterende map være meget tæt på linien  $y = x$ . Med andre ord, er det en god forudsigelse at hævde, at værdien af tidsserien til tid  $t + \tau$ , hvor  $\tau$  er lille, er den samme som værdien til tid  $t$ .

Dette kaldes *persistens*. Begynder man at gøre  $\tau$  større, det vil sige ønsker man at forudsige længere frem i tiden, bliver map'en mere kompliceret, og det er ikke muligt at slippe så "billigt" fra forudsigelsen. For tidsserie C er der ingen forskel på netværk 1a og 1b.

Da det er map'en netværket optrænes i, har det den konsekvens, at selv det simple netværk 1a kan oplæres i og forudsige en funktion, der, som tidsserie betragtet, ser meget kaotisk ud, men har en simpel map. Kigger vi på tidsserie D, se Figur 15, der også kaldes *Feigenbaum* serien, kan man vise, at den er en kaotisk og ergodisk<sup>14</sup> tidsserie. Det kaos, der er tale om, er selvfølgelig deterministisk kaos. At et system er kaotisk, betyder at det er (meget) følsomt overfor startbetingelserne - det vil sige en lille perturbation medfører en meget større effekt på et senere tidspunkt<sup>15</sup>. Tidsserie D er givet ved rekursionsformlen:

$$z_{t+1} = 4z_t(1 - z_t), \quad z_0 = 0.1$$

Ser man på seriens map, viser det sig imidlertid, at den er meget simpel, se Figur 16.

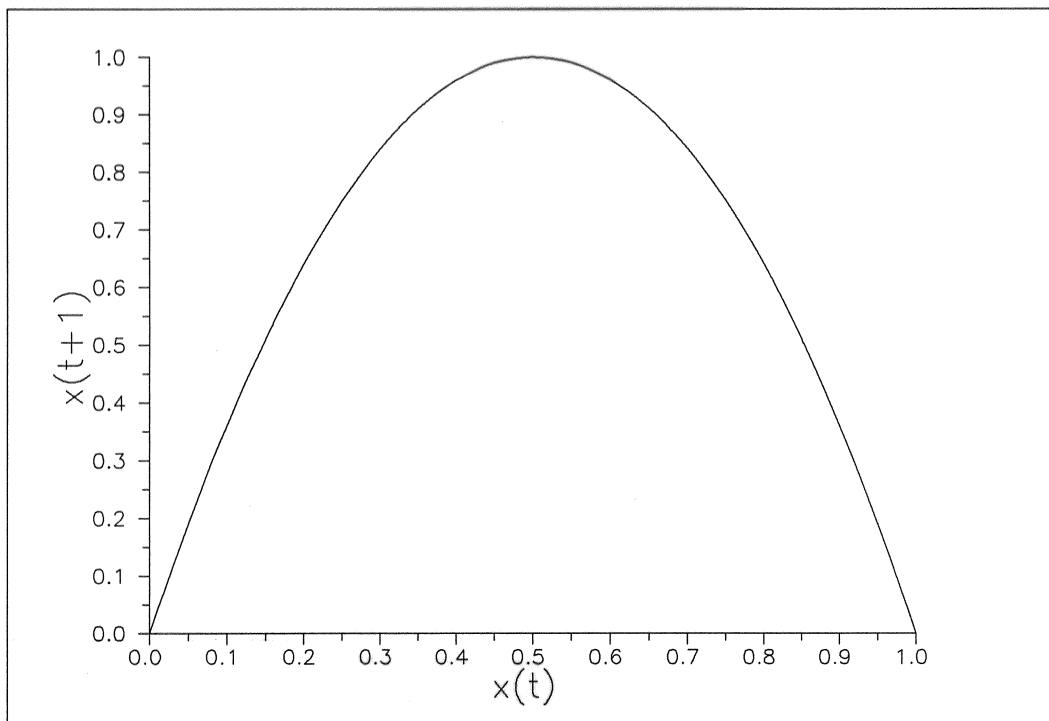


Figur 15. Tidsserie D.

Grunden til, at denne tidsserie i det hele taget er med, er at mange (for eksempel Lapedes and Farber (1987) og Farmer and Sidorowich, 1988) mener, at meget af det, der ser *tilfældigt* ud i naturen, faktisk udviser deterministisk kaotisk opførsel. Indtil for nylig har man ment, at tilfældighed var forårsaget af ekstrem komplikation, det vil sige mange frihedsgrader. Idag betragter man tilfældighed som skabt af kaos, der kraftigt forstærker det, vi ikke ved om startbetingelserne. Så måske kan noget af en temperaturseries "tilfældighed" forklares på denne måde. Deterministisk kaos er fundet i mange fysiske situationer, for eksempel ved overgangen til turbulent strømning i væsker (Ruelle and Takens, 1971, Swinney, 1978),

<sup>14</sup>Ergodisk betyder at middelværdier af alle mulige størrelser, dannet ud fra tidsserien, konvergerer.

<sup>15</sup>Generelt om kaos, se for eksempel Schuster (1984).



Figur 16. Map'en af tidsserie D.

kemiske reaktioner (Tomita et al., 1979), lasere (Haken, 1975) og i plasma fysik (Russell, 1980). Derfor er det vigtigt at se, om netværkene kan optrænes i disse maps. Endvidere hævdes det, at disse fænomeners *map* ofte kolliderer til en simpel map.

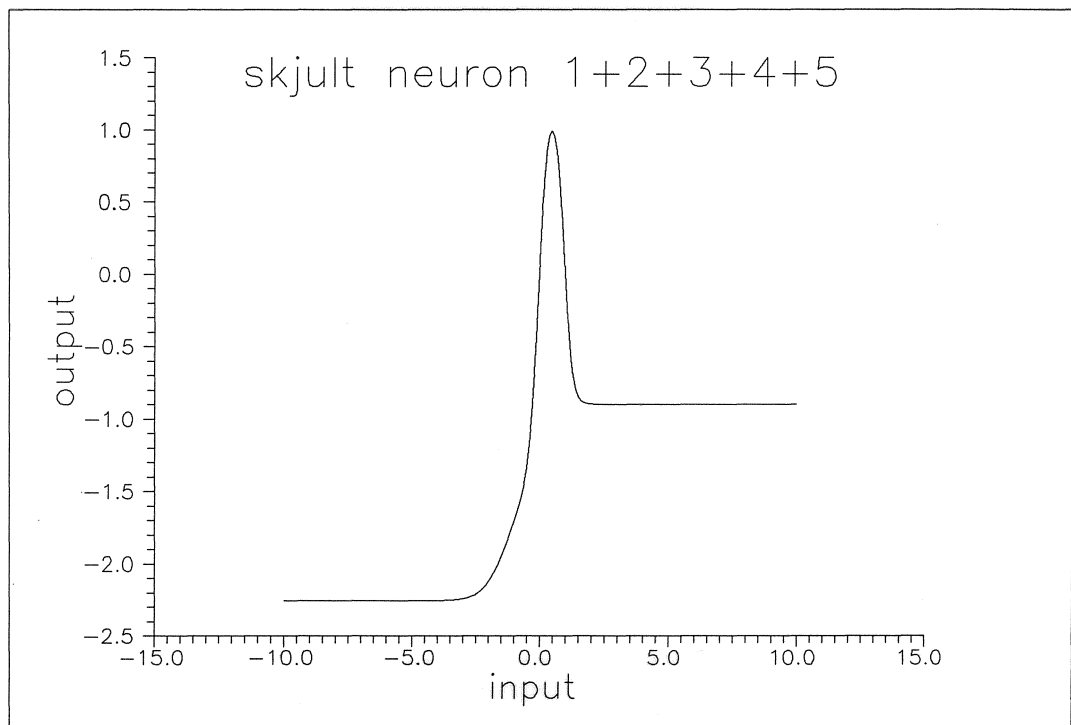
Ser man på, hvordan netværk 1a har lært tidsserie D, ses det, at det har lært serien godt, se Tabel 2. Netværkets responsfunktion er vist i Figur 17. Af denne figur fremgår det, at responsfunktionen taget over et større område igen har form som aktiveringsfunktionen, undtagen i intervallet  $[0;1]$ .

Man taler ofte om, at neurale netværk besidder evnen til at *generalisere*. Ser vi igen på Figur 17, kan man se, at nettet faktisk i intervallet  $[0;1]$  har generaliseret, i den forstand at det har lavet en kontinuert kurve mellem de punkter, det er blevet optrænet i - selvom nettet er blevet præsenteret for punkterne i helt tilfældig rækkefølge. Det kan derimod ikke siges, at nettet har generaliseret *udenfor* det område, det er blevet optrænet i. I dette tilfælde er det også svært (=umuligt), for værdimængden af Feigenbaum funktionen er netop intervallet  $[0;1]$ . Der er ingen forskel på netværk 1a og 1b.

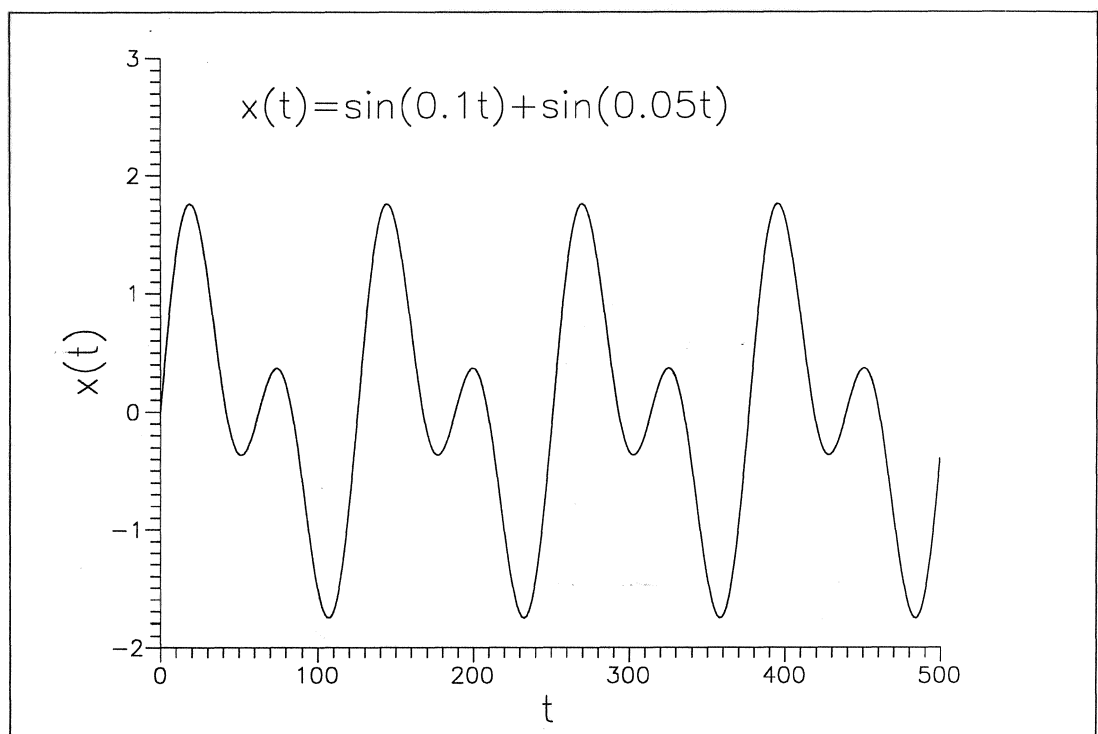
Efter denne simple map (men komplicerede tidsserie) fortsættes med de komplicerede *map'er*. Netværk 1a optrænes i tidsserie E, givet ved:

$$z_t = \sin(0.1t) + \sin(0.05t).$$

selve tidsserien er vist i Figur 18 og dens meget smukke map i Figur 19. Denne map er yderst kompliceret; idet der til én værdi af  $z_t$  hører flere (op til 4) senere værdier,  $z_{t+1}$ . Ser man på, hvordan det er gået, ses det at netværk 1a faktisk har lært map'en meget godt, se Tabel 2. Forklaringen kunne være, at grafen af map'en ligger omkring linien  $x = y$ , og at ellipsen stadig er så fladtrykt, at fejlen ved at approksimere map'en med linien  $x = y$  er lille, ligesom for tidsserie C. Bemærk dog, at både fejlen og spredningen er større end for de simple tidsserier. Der er igen ingen forskel på 1a og b.



Figur 17. Netværk 1a's responsfunktion efter optræning i tidsserie D.

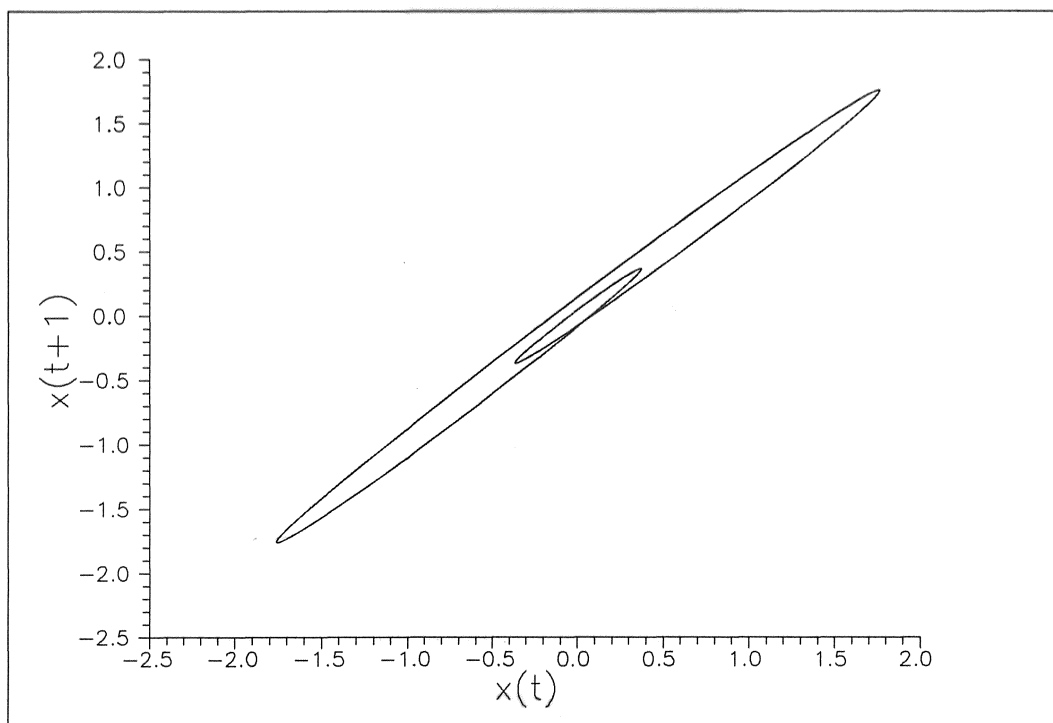


Figur 18. Tidsserie E.

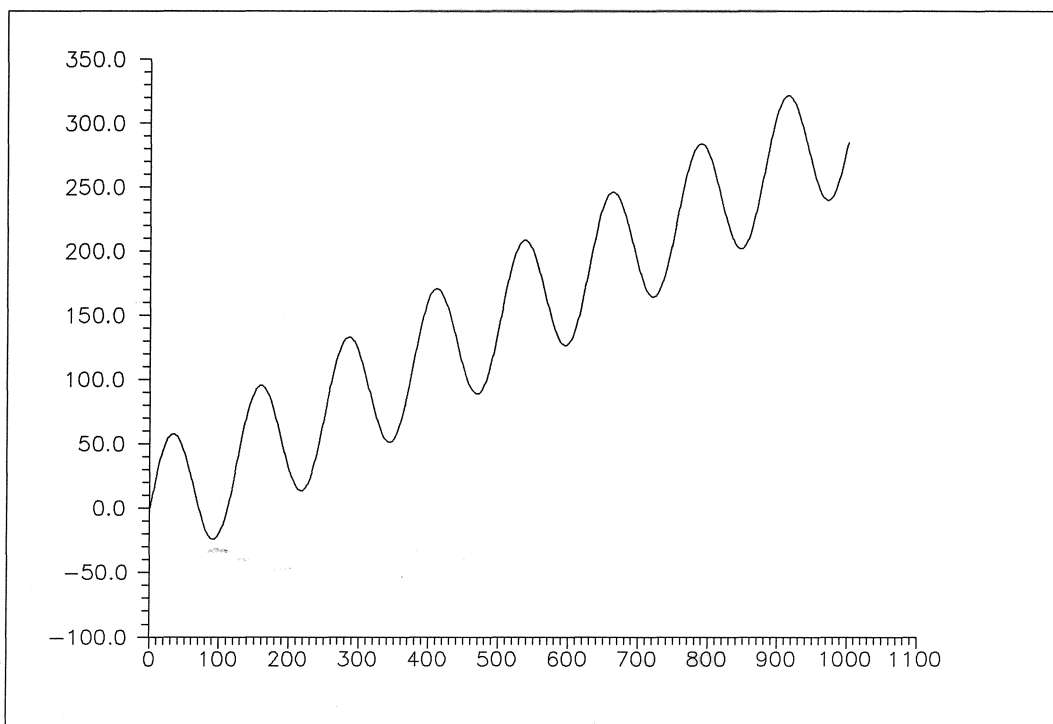
Den sidste tidsserie, vi prøver at lære netværk 1a, er serien F, givet ved:

$$z_t = 0.3t - 2 + 50 \sin(0.05t).$$

På Figur 20 er selve tidsserien tegnet og map'en kan ses på Figur 21. Som det ses, er tidsserien en kombination af en periodisk svingning og lineær vækst. Map'en



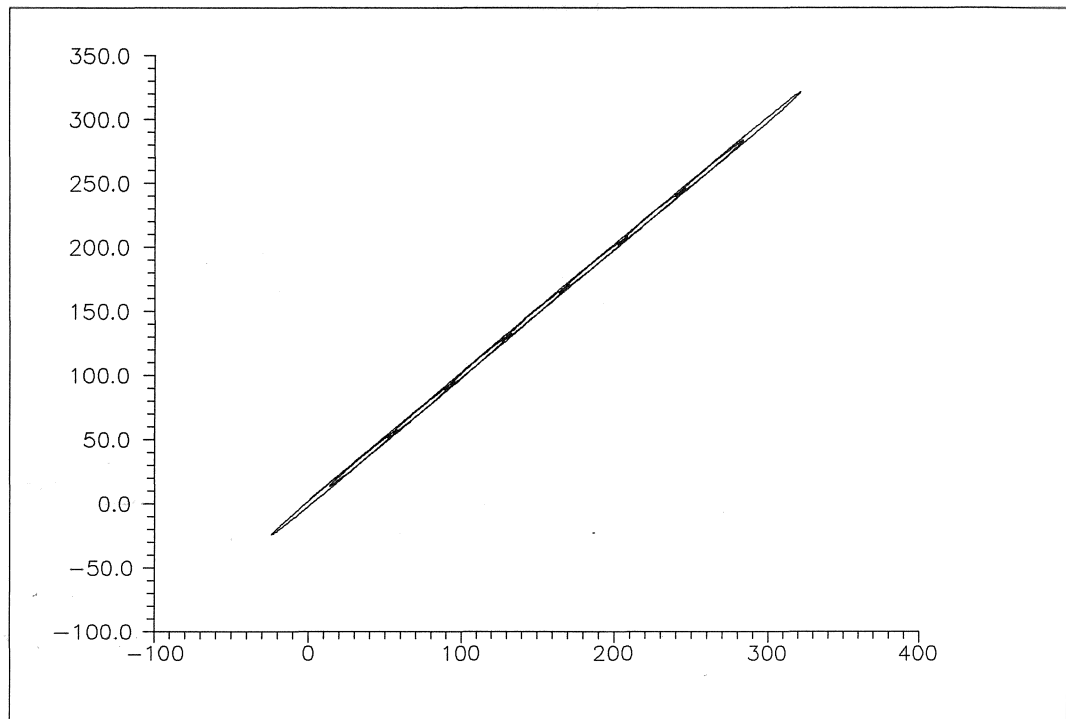
Figur 19. Map'en af tidsserie E.



Figur 20. Tidsserie F.

får et meget flurligt udseende, idet den er en udtrukket variation af map'en for tidsserie F. Igen er den centreret om linien  $x = y$ . Som for tidsserie B er det ikke lykkedes særligt godt med oplæringen, det er det derimod for netværk 1b.





Figur 21. Map'en af tidsserie F.

Generelt er forskellen mellem 1a og 1b, at de tidsserier, der har en lineær opførsel (det vil sige A, B og F) er lært meget bedre af 1b, se Tabel 2. Dette bør ikke komme bag på nogen, da den direkte forbindelse fra inputneuronen til outputneuronen netop forsyner nettet med lineære egenskaber. Konklusionen på dette er, at hvis man også vil gøre sig forhåbninger om at kunne håndtere lineære fænomener, så må denne direkte forbindelse inkluderes i nettet.

For *netværk 2a+b* gælder det, at der er langt flere skjulte neuroner, samt 2 lag af dem. Dette betyder at nettet – alt andet lige – får en større opløsningsevne. Det betyder samtidigt, at nettet skal trænes længere, det vil sige med flere præsentationer for at opnå den samme nøjagtighed som netværk 1a+b. Lapedes og Farber (1987) har dog fundet, at map'er kan blive så kompliceret, at netværk 1 ikke er i stand til at lære dem tilfredsstillende. Det er dog værd at bemærke, at med den nøjagtighed, der er valgt i disse eksperimenter, har der ikke været nogle tidsserier, der har været så komplicerede, at netværk 1 ikke har kunnet lære dem. Kigger man imidlertid på den relative fejl,  $r$ , ses det at netværk 2b er det eneste, der har en lille relativ fejl for *alle* eksperimenterne, se Table 2. Der er to grunde til, at de andre net har så stor en relativ fejl:

1. For netværk 1a og 2a og tidsserie B og F's vedkommende skyldes det, at netværkene simpelthen ikke har lært tidsserierne. Dette afspejler sig også i de andre størrelser i tabellen.
2. For netværk 1a og 1b og tidsserie D skyldes det, at tidsserien har værdier, der er meget små (af størrelsesordenen  $10^{-8}$ ) og netværket prognosticerer en relativt for stor værdi (af størrelsesordenen  $10^{-2}$ ), dette giver netop en meget stor relativ fejl (af størrelsesordenen  $10^6$ ), der helt kommer til at dominere den totale relative fejl.

Den forholdsvis store relative fejl for netværk 1b og tidsserie E kan forklares ved, at nettet formentlig har lagt for stor vægt på den direkte forbindelse mellem input-

og outputneuronen. Således at det bliver  $x = y$ , der dominerer nettets responsfunktion. Som for netværk 1a har netværk 2a også svært ved at lære map'er, der har lineære egenskaber, dette klares dog - ligesom af 1b - også af 2b.

## 4.4 Andre netværk til forudsigelse af tidsserier

Det netværk, der er anvendt i denne rapport har visse fordele og visse ulemper. En fordel er, at den type netværk, der er anvendt her, er et meget generelt netværk; det kan ligeså godt anvendes til at forudsige proteiners 3-dimensionale struktur, se Bohr et al. (1988), som det kan anvendes til orddeling. Så som en introduktion til neurale netværk er det meget egnet, på grund af dets generalitet. En bagdel ved det er, at det lærer langsomt, men giver et hurtigt svar, når det er oplært - ligesom mennesket.

Det er muligt at komme denne langsommelighed til livs på flere forskellige måder. Inden for rammerne af den valgte arkitektur kan man forsøge at optimere minimeringsprocessen, ved at anvende andre metoder end "steepest descent", for eksempel "conjugated gradient descent" eller "stochastic gradient descent", se Af-snit 3.

Moody (Moody and Darken, 1989 og Moody, 1989.) har udviklet en helt anden arkitektur og oplæringsalgoritme for netværk, der for eksempel skal anvendes til at forudsige tidsserier. Ideen er, at netværket sættes sammen af neuroner, der kun reagerer på input fra et vist udsnit af inputrummet, disse kaldes for "*locally tuned*" neuroner. Responsfunktionen for den  $\alpha$ 'te locally tuned neuron er givet ved

$$R^\alpha(\vec{x}) = R(|\vec{x} - \vec{x}^\alpha|/\sigma^\alpha)$$

hvor  $R$  er en radiært symmetrisk funktion med et enkelt maximum i nul, der hurtigt falder af mod nul for store radier,  $\vec{x}$  er inputvektoren og  $\vec{x}^\alpha$  og  $\sigma^\alpha$  er henholdsvis centrum og bredden i inputrummet af den  $\alpha$ 'te enhed. Som  $R$  anvender Moody en normaliseret gaussisk responsfunktion givet ved

$$R^\alpha(\vec{x}) = \exp \left[ -\frac{|\vec{x} - \vec{x}^\alpha|^2}{(\sigma^\alpha)^2} \right]$$

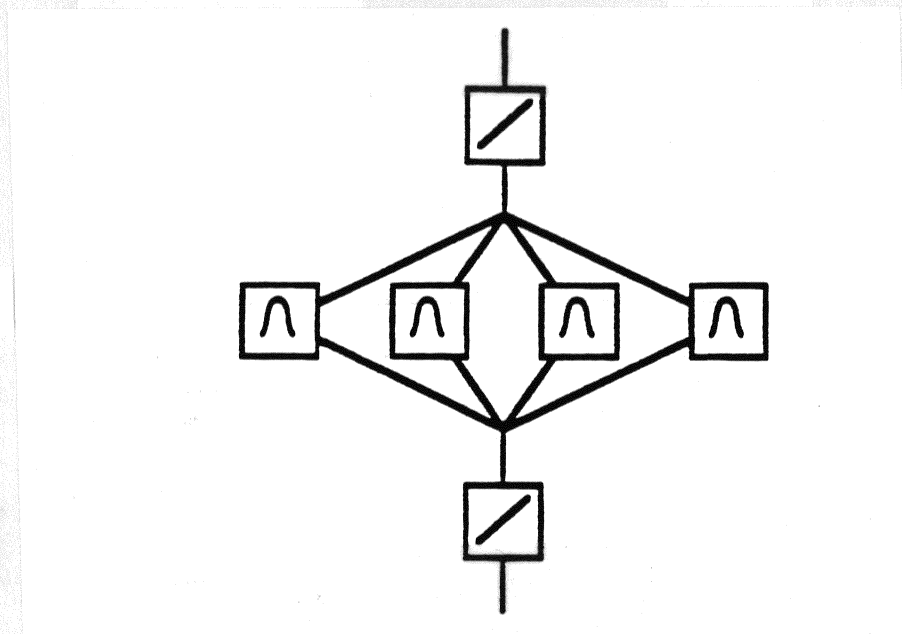
Et eksempel på et netværk, der består af sådanne locally tuned neuroner, er vist i Figur 22. Output'et fra et netværk, der består af  $M$  sådanne enheder, er givet ved

$$f(\vec{x}) = \sum_{\alpha=1}^M A^\alpha R^\alpha(\vec{x})$$

hvor  $A^\alpha$  er amplituden af den  $\alpha$ 'te enhed. Optrænes nettet ved at minimere den totale fejl,  $E = \frac{1}{2} \sum_{i=1}^N (f^*(\vec{x}_i) - f(\vec{x}_i))^2$ , hvor  $N$  er antallet af input/output-par og  $f^*$  er det ønskede output, ved hjælp af steepest descent, finder man at denne metode er ligeså præcis (og ligeså langsom) som almindelig back-propagation. Samtidig viser det sig, at bredden af de enkelte responsfunktioner bliver så stor, at de ikke kan betragtes som lokale længere. Moody har derfor udviklet en *hybrid* oplæringsregel, der består i først at bestemme centrene af de enkelte responsfunktioner ved en selvorganiserende metode, der minimerer den Euklidiske afstand mellem de  $N$  input/output-par og centret af  $k$  af de  $M$  enheder, der er nærmest, dette er "k-means clustering" algoritmen. Når centrene er fastlagt, bestemmes bredden af hver responsfunktion ved at variere den, for at opnå at den enkelte responsfunktion overlapper naboresponsfunktionen, således at inputrummet bliver interpoleret tilfredsstillende i det område responsfunktionen dækker. Siden bliver amplituderne,  $A^\alpha$ , varieret således, at de minimerer den totale fejl,  $E$ . Moody har fundet, at denne sidste del konvergerer meget hurtigt, da meget af arbejdet er gjort på forhånd. Anvendes denne metode til at forudsige løsninger af Machey-Glass differentiaalligningen (der er et "svært" problem) og sammenlignes beregningstiden



(præcisionen er den samme) med traditionel back-propagation, finder man, at Moody's metode er 100 til 1,000 gange hurtigere !



*Figur 22. Et neuralt netværk bestående af locally-tuned neuroner. Fra Moody and Darken (1989).*

## **Part II**

# **ARIMA processen**



## 5 Teori

### 5.1 Indledning

Anden del af denne rapport handler, som nævnt i indledningen, om ARIMA-processen. ARIMA er forkortelsen for AutoRegressive Integrated Moving Average, og er en lineær stokastisk metode, der består i at modellere tidsserien v.h.j.a. de 2 i forkortelsen nævnte stokastiske processer. Processerne er:

1. Autoregression
2. Glidende midling

Begge processer behøver ikke nødvendigvis at indgå i modellen. Der er ikke tale om en deterministisk model. Dette betyder, at det ikke er ud fra fysiske overvejelser, modellen laves, men derimod udelukkende ud fra selve tidsseriens udseende. En stokastisk proces er en proces, hvor der til et endeligt sandsynlighedsfelt,  $(U, P)$ , er knyttet en afbildning af udfaldsrummet,  $U$ , ind i de reelle tal,  $R$ . Denne afbildning kaldes en *stokastisk variabel*.  $P$  kaldes sandsynlighedsfunktionen. Meget kort kan dette resumeres som: *En tilfældig årsag har et reelt tal som konsekvens*. Når modellen er lavet, bruges den til at lave forudsigelser med. I dette afsnit gennemgås teorien for ARIMA-processen. "Grundbogen" om ARIMA-processer er Box og Jenkins' bog (Box and Jenkins, 1970). Det er den, der ligger til grund for det efterfølgende.

ARIMA-modellen er også blevet anvendt på en anden temperaturserie af Petersen og Larsen (1978). De brugte 700,000 års temperaturdata til at forsøge at udvikle en model, der simulerer den klimatiske variation. Deres konklusion var at ARIMA-modellen, fra et statistisk synspunkt, har svært ved at klare forudsigelser langt frem i tiden, men "one-step-ahead" forudsigelser – hvilket er den type af forudsigelser der anvendes hér – klarer den godt.

## 6 Autokovarians- og autokorrelationskoefficienten

Det vigtigste værktøj i bestemmelsen af hvilken model, der skal anvendes, er autokorrelationskoefficienten. Denne er beskrevet i dette afsnit. Et andet hyppigt anvendt værktøj er den partielle autokorrelationskoefficient, der vil blive beskrevet senere.

### Autokorrelationskoefficienten

En stokastisk proces kaldes *strengt stationær* eller blot *stationær*, hvis dens (statistiske) egenskaber er uforandrede, når tidsaksens begyndelsespunkt flyttes. Dette betyder, at den fælles (engelsk: joint) sandsynlighedsfordeling for  $m$  observationer  $z_{t_1}, z_{t_2}, \dots, z_{t_m}$ , til de *ækvivalente* tider  $t_1, t_2, \dots, t_m$ , er den samme som for  $m$  observationer  $z_{t_1+k}, z_{t_2+k}, \dots, z_{t_m+k}$  til tiderne  $t_1 + k, t_2 + k, \dots, t_m + k$ . Gælder der for processen, at  $m = 1$ , er dens sandsynlighedsfordeling  $p(z_t)$  den samme for alle  $t$ , og sandsynlighedsfordelingen kan derfor skrives  $p(z)$ . *Middelværdien* for denne proces defineres da som

$$\mu = E[z_t] = \int_{-\infty}^{\infty} zp(z)dz$$

og dens *varians* som

$$\sigma_z^2 = E[(z_t - \mu)^2] = \int_{-\infty}^{\infty} (z - \mu)^2 p(z) dz$$

For en tidsserie, der består af diskrete målinger, estimeres  $\mu$  ved

$$\bar{z} = \frac{1}{N} \sum_{t=1}^N z_t$$

hvor  $N$  er antallet af målinger. Variansen estimeres ud fra

$$\hat{\sigma}_z^2 = \frac{1}{N} \sum_{t=1}^N (z_t - \bar{z})^2$$

Man indfører nu begrebet *autokovarians med forsinkelsen (lag)  $k$* , der er defineret ved

$$\gamma_k = \text{cov}[z_t, z_{t+k}] = E[(z_t - \mu)(z_{t+k} - \mu)] \quad (5)$$

og tilsvarende *autokorrelationen med forsinkelsen (lag)  $k$*

$$\rho_k = \frac{E[(z_t - \mu)(z_{t+k} - \mu)]}{\sqrt{E[(z_t - \mu)^2]E[(z_{t+k} - \mu)^2]}} = \frac{E[(z_t - \mu)(z_{t+k} - \mu)]}{\sigma_z^2}$$

da der for en stationær proces, gælder at variansen,  $\sigma_z^2 = \gamma_0$ , er den samme til tid  $t+k$  som til tid  $t$ . Dette medfører, at autokorrelationen med forsinkelse  $k$  kan skrives

$$\rho_k = \frac{\gamma_k}{\gamma_0}$$

af dette ses, at  $\rho_0 = 1$ . Plottet af  $\gamma_k$  mod  $k$  kaldes *autokovariansfunktionen*  $\{\gamma_k\}$ , og plottet af  $\rho_k$  mod  $k$  kaldes *autokorrelationsfunktionen*  $\{\rho_k\}$ .

I praksis har man kun en endelig tidsserie, og man kan derfor kun bestemme estimater af autokorrelationerne. Statistikere har forsøgt at estimere dem på mange måder (se for eksempel Jenkins and Watts, 1968), og de er kommet frem til, at det bedste estimat - og det mest indlysende - af  $\rho_k$  er

$$r_k = \frac{c_k}{c_0}$$

hvor

$$c_k = \frac{1}{N} \sum_{t=1}^{N-k} (z_t - \bar{z})(z_{t+k} - \bar{z}), \quad k = 0, 1, 2, \dots, K$$

er estimatet af autokovariansen  $\gamma_k$  og  $K < N$ ,  $c_k$  og dermed  $r_k$  har kun mening for  $K$  lig cirka  $N/2$ , da der for større værdier indgår for få tal i beregningen.

Da det er estimater, der beregnes, er der nødvendigvis en fejl forbundet med dem. Bartlett (1946) har fundet, at en approksimation af variansen på estimatet af autokorrelationen er

$$\text{var}[r_k] \simeq \frac{1}{N} \sum_{v=-\infty}^{\infty} (\rho_v^2 + \rho_{v+k}\rho_{v-k} - 4\rho_k\rho_v\rho_{v-k} + 2\rho_v^2\rho_k^2)$$

Gælder der, at den *teoretiske* autokorrelation er nul for  $k > q$ , bliver

$$\text{var}[r_k] \simeq \frac{1}{N} \{1 + 2 \sum_{v=1}^q \rho_v^2\}, \quad k > q \quad (6)$$



## Autokorrelationsmatricen

Kovariansmatricen hørende til en stationær proces  $(z_1, z_2, \dots, z_n)$  er givet ved

$$\begin{aligned}\Gamma_n &= \begin{bmatrix} \gamma_0 & \gamma_1 & \gamma_2 & \cdots & \gamma_{n-1} \\ \gamma_1 & \gamma_0 & \gamma_1 & \cdots & \gamma_{n-2} \\ \gamma_2 & \gamma_1 & \gamma_0 & \cdots & \gamma_{n-3} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \gamma_{n-1} & \gamma_{n-2} & \gamma_{n-3} & \cdots & \gamma_0 \end{bmatrix} \\ &= \sigma_z^2 \begin{bmatrix} 1 & \rho_1 & \rho_2 & \cdots & \rho_{n-1} \\ \rho_1 & 1 & \rho_1 & \cdots & \rho_{n-2} \\ \rho_2 & \rho_1 & 1 & \cdots & \rho_{n-3} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \rho_{n-1} & \rho_{n-2} & \rho_{n-3} & \cdots & 1 \end{bmatrix} = \sigma_z^2 \mathbf{P}_n\end{aligned}$$

Kovariansmatricer, der er beskaffen som  $\Gamma_n$ , det vil sige matricer, der er symmetriske med konstante elementer på alle diagonaler, kaldes for *autokovariansmatricer*, og de tilhørende matricer  $\mathbf{P}_n$  kaldes *autokorrelationsmatricer*. Betragt nu en vilkårlig linear kombination af  $z_t, z_{t-1}, \dots, z_{t-n+1}$

$$L_t = l_1 z_t + l_2 z_{t-1} + \cdots + l_n z_{t-n+1}$$

da  $\text{cov}[z_i, z_j] = \gamma_{|j-i|}$  for en stationær proces, fordi det kun er deres indbyrdes afstand, der spiller en rolle, da deres placering ikke kan have betydning, fordi nulpunktet kan placeres vilkårligt, er variansen af  $L_t$  givet ved

$$\text{var}[L_t] = \sum_{i=1}^n \sum_{j=1}^n l_i l_j \gamma_{|i-j|}$$

der er positiv, hvis ikke alle  $l$ 'erne er nul, fordi variansen altid er positiv. Er alle  $l$ 'er nul, er variansen også nul. Dette er netop definitionen på en positiv definit matrix, så både autokovarians- og autokorrelationsmatricen for en stationær proces er *positivt definitte*<sup>16</sup>.

## 6.1 Om temperaturserien

I dette afsnit vil der blive foretaget en "klassisk" analyse af tidsserien. Tidsserien består af temperaturen observeret på Fanø, nærmere bestemt i Nordby. Som nævnt i indledningen er den første måling foretaget den 1. januar 1875 og den sidste den 9. maj 1969. Alle målingerne er foretaget kl. 8 om morgenen lokal tid. Tidsserien er et uddrag af en større pakke af meteorologiske data fra Fanø. Pakken består foruden af temperaturen observeret kl. 8, også af lufttryk kl. 8, daglig maximum- og minimumtemperatur samt af tryk og temperatur målt kl. 14 og 21. Pakken er blevet analyseret af mange: E. W. Peterson og S. E. Larsen (Peterson og Larsen, 1983 og Peterson og Larsen, 1984), der har beskrevet 30 års løbende midler af vind, temperatur, tryk, nedbør og skydække. N. Woetmann Nielsen og A. Walløe Hansen (Nielsen og Hansen, 1986), der har studeret den årlige variation af temperatur og tryk. Endvidere har N. Woetmann Nielsen analyseret den indre årlige variation i tryk- og temperaturfordelingens standardafvigelse og skævhedskoefficient (Nielsen, 1988).

Temperaturseriens ekstremer, gennemsnit og spredning kan ses i Tabel 3.

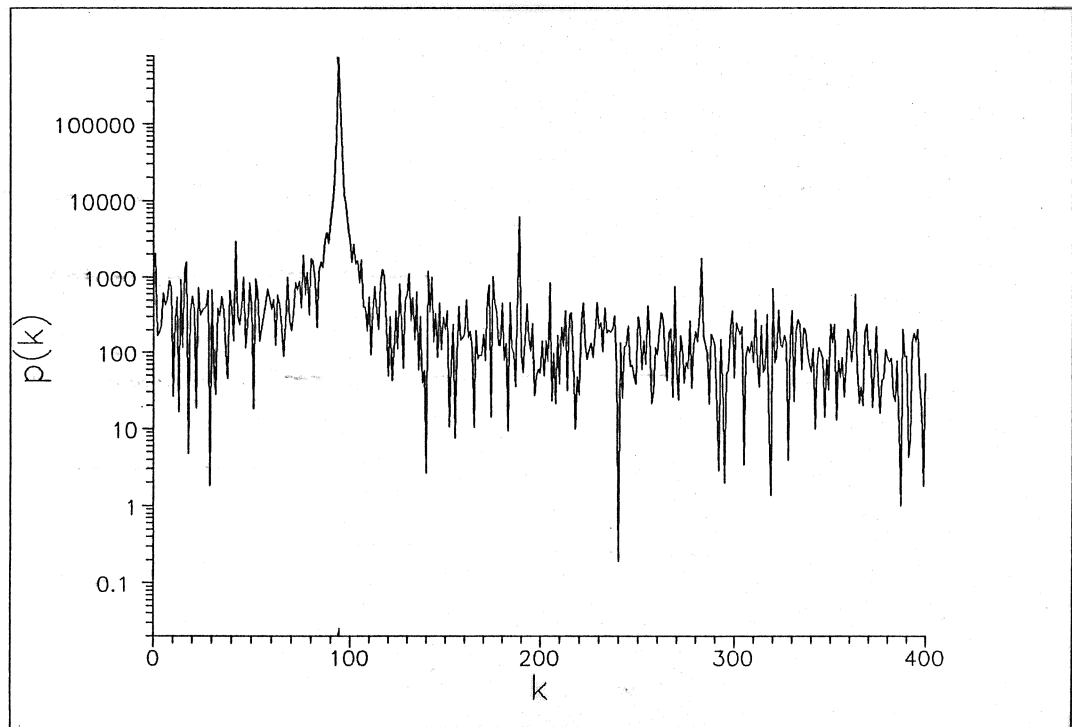
### Spektrale egenskaber

På Figur 23 ses temperaturseriens periodogram. Periodogrammet, der blev indført

<sup>16</sup>En anden definition, af at en matrix er *positivt definit*, er, at dens rang og index er ens. Om rang og index se for eksempel Noter til Mat A (1978) og Ayres (1962).

Tabel 3. Temperaturseriens ekstremer, gennemsnit og spredning.

Minimum	-22.0°C
Maximum	26.7°C
Gennemsnit	7.60°C
Spredning	6.63°C



Figur 23. Temperaturseriens periodogram.  $f(k) = f_k = k/N$  er den  $k$ 'te frekvens og  $N=34,463$  er antallet af observationer.

af Schuster allerede i 1898 (se Schuster, 1898), siger noget om en tidsseries spektrale egenskaber. Det består af:

$$I(f_i) = \frac{N}{2}(a_i^2 + b_i^2), \quad i = 1, 2, \dots, q$$

Hvor  $I(f_i)$  kaldes *intensiteten* og er et udtryk for hvor kraftig en bestemt frekvens,  $f_i = i/N$ , er.  $N = 2q + 1$  er antallet af observationer, for temperaturserien er  $N = 34,463$ ,

$$a_i = \frac{2}{N} \sum_{t=1}^N z_t c_{it}$$

og

$$b_i = \frac{2}{N} \sum_{t=1}^N z_t s_{it}$$

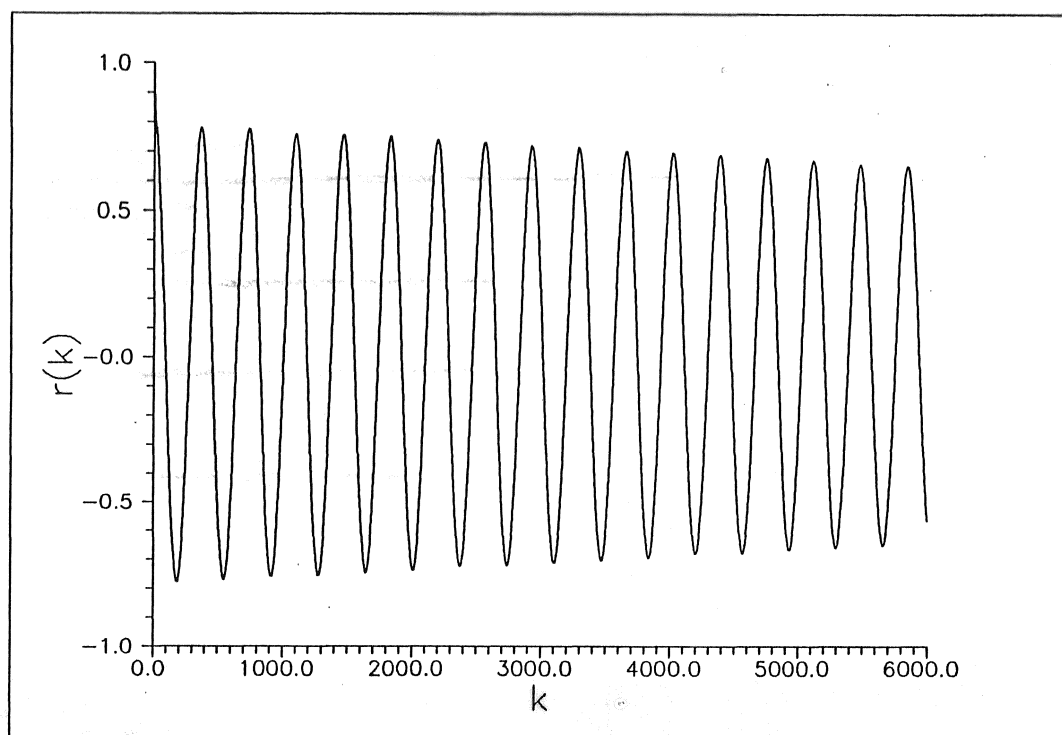
hvor  $c_{it} = \cos 2\pi f_i t$  og  $s_{it} = \sin 2\pi f_i t$ .  $a_i$  og  $b_i$  er mindste kvadraters fit til koefficienterne i den trunkerede Fourierserie's  $i$ 'te harmoniske med  $1/N = 2.90 \cdot 10^{-5}$  som fundamental frekvens<sup>17</sup>. Af plottet ses, at der er en altdominerende peak ved den 94de harmoniske (bemærk at y-aksen er logaritmisk), hvilket svarer til en periode på  $34,463/94=365$  dage, der netop er den, på grund af Jordens bevægelse

<sup>17</sup>For et bevis af dette, se for eksempel Scheid (1968).

om Solen, forventede årlige svingning. Omkring 190 ses den 1. oversvingning og omkring 285 den 2. Den fjerde største intensitet er omkring 45, dette er en 2-årlig svingning.

### Autokorrelationskoefficienterne

På Figur 24 ses tidsseriens autokorrelationskoefficienter. Det ses, at der er en dæmpet svingning med periode  $k = 365$  døgn. Årsagen er igen den årlige svingning. Dæmpningen skyldes, at jo større lag'et er, jo mere indflydelse får støjen i signalet.



Figur 24. Plot af temperaturseriens autokorrelationsfunktion

## 6.2 Stationære modeller

I dette afsnit beskrives de to processer, der indgår i ARIMA-processen, en beskrivelse af ARIMA-processen selv og nogle betingelser, der skal være opfyldt for enhver ARIMA-proces.

### Autoregressive processer

I den *autoregressive* model udtrykkes den nuværende værdi af tidsserien ved en endelig, lineær sum af tidligere værdier og et "chok",  $a_t$ . Disse chok er normalfordelte med gennemsnit 0 og varians  $\sigma_a^2$ . Processen, der frembringer  $a_t$ 'erne, kaldes for *hvid støj*. Værdien af tidsserien til tiden  $t, t-1, t-2, \dots$  skrives  $z_t, z_{t-1}, z_{t-2}, \dots$ . Endvidere kaldes forskellen mellem  $z_t$  og seriens middelværdi,  $\mu$ , for  $\tilde{z}_t$ . Modellen kan altså skrives

$$\tilde{z}_t = \phi_1 \tilde{z}_{t-1} + \phi_2 \tilde{z}_{t-2} + \dots + \phi_p \tilde{z}_{t-p} + a_t. \quad (7)$$

Denne model kaldes for en *autoregressiv proces af orden p* og skrives kort  $AR(p)$ . Man indfører en *autoregressiv operator* af orden  $p$ ,  $\phi(B)$ , defineret ved

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p$$

hvor  $B$  er *backward shift* operatoren defineret ved

$$B^m z_t = z_{t-m}$$

specielt er  $Bz_t = z_{t-1}$ . Ved brug af den autoregressive operator kan modellen skrives

$$\phi(B)\tilde{z}_t = a_t$$

Denne model indeholder  $p+2$  parametre  $\mu, \phi_1, \phi_2, \dots, \phi_p, \sigma_a^2$ , der skal bestemmes ud fra data.

Af Ligning (7) ses det, at  $\tilde{z}_{t-1}$  kan udtrykkes ved

$$\tilde{z}_{t-1} = \phi_1 \tilde{z}_{t-2} + \phi_2 \tilde{z}_{t-3} + \dots + \phi_p \tilde{z}_{t-p-1} + a_{t-1}$$

Ligeledes for  $\tilde{z}_{t-2}$  og så videre, hvilket fører til en *uendelig* serie af  $a$ 'er. Det vil sige, at

$$\phi(B)\tilde{z}_t = a_t$$

kan skrives

$$\tilde{z}_t = \psi(B)a_t$$

hvor  $\psi(B) = \phi^{-1}(B)$  er en operator med uendeligt mange led.

**Autokorrelationsfunktionen for en AR( $p$ )-proces** En vigtig rekursiv formel til at beregne autokorrelationerne for en autoregressiv proces af orden  $p$ , fås ved at gange igennem med  $\tilde{z}_{t-k}$  i Ligning (7), dette giver

$$\tilde{z}_{t-k}\tilde{z}_t = \phi_1 \tilde{z}_{t-k}\tilde{z}_{t-1} + \phi_2 \tilde{z}_{t-k}\tilde{z}_{t-2} + \dots + \phi_p \tilde{z}_{t-k}\tilde{z}_{t-p} + \tilde{z}_{t-k}a_t$$

Midt over  $t$  og anvendes definitionen på autokovarianskoefficienten, se Ligning (5), fås følgende differensligning

$$\gamma_k = \phi_1 \gamma_{k-1} + \phi_2 \gamma_{k-2} + \dots + \phi_p \gamma_{k-p}, \quad k > 0$$

Bemærk, at  $E[\tilde{z}_{t-k}a_t]$  er nul for  $k > 0$ , da  $\tilde{z}_{t-k}$  kun indeholder chok op til og med tid  $t-k$ , der ikke er korrelerede med  $a$  til tid  $t$ . Divideres med  $\gamma_0$  fås

$$\rho_k = \phi_1 \rho_{k-1} + \phi_2 \rho_{k-2} + \dots + \phi_p \rho_{k-p}, \quad k > 0 \quad (8)$$

Skrives dette ved hjælp af den autoregressive operator, fås

$$\phi(B)\rho_k = 0$$

hvor  $\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p$  og  $B$  nu virker på  $k$  og ikke som normalt  $t$ . Skrives

$$\phi(B) = \prod_{i=1}^p (1 - G_i B)$$

er den generelle løsning til (8)

$$\rho_k = A_1 G_1^k + A_2 G_2^k + \dots + A_p G_p^k \quad (9)$$

hvor  $G_1^{-1}, G_2^{-1}, \dots, G_p^{-1}$  er rødder i den *karaktæriske ligning*

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p = 0$$

Stationaritet, se Afsnit 6.2, kræver at  $|G_i| < 1$ . Dette medfører - idet vi antager, at  $G_i$ 'erne er forskellige - at to situationer kan opstå:

1. En rod  $G_i$  er reel, således at leddet  $A_i G_i^k$  i Ligning (9) aftager eksponentielt mod nul, når  $k$  vokser.
2. Et par af rødderne  $G_i, G_j$  er komplekse, således at de bidrager med størrelsen

$$A d^k \sin(2\pi f k + F), \quad d < 1$$

i Ligning (9). Dette er en dæmpet sinusfunktion.

Generelt vil autokorrelationsfunktionen for en autoregressiv proces bestå af en blanding af både dæmpede eksponential- og sinusfunktioner.



Indsættes  $k = 1, 2, \dots, p$  i Ligning (8) fås



Denne model indeholder  $q + 2$  parametre  $\mu, \theta_1, \dots, \theta_q, \sigma_a^2$ , der skal bestemmes ud fra data.

Analogt med AR-processen kan udtrykket for  $a_{t-1}$  skrives

$$a_{t-1} = \tilde{z}_{t-1} + \theta_1 a_{t-2} + \theta_2 a_{t-3} + \dots + \theta_q a_{t-q-1}$$

tilsvarende for  $a_{t-2}$  og så videre, hvilket fører til at  $a_t$  kan udtrykkes ved en uendelig sum af  $\tilde{z}_t$ 'er

$$a_t = \pi(B)\tilde{z}_t$$

hvor  $\pi(B) = \theta^{-1}(B)$ .

**Autokorrelationer for en glidende midlings proces** Anvendes definitionen på autokorrelationen, og indsættes udtrykket for  $\tilde{z}_t$ , se Ligning (10), i denne, fås at autokorrelationen for en glidende midlings proces kan skrives

$$\gamma_k = E[(a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q})(a_{t-k} - \theta_1 a_{t-k-1} - \dots - \theta_q a_{t-k-q})]$$

Dette betyder, at variansen er

$$\gamma_0 = (1 + \theta_1^2 + \theta_2^2 + \dots + \theta_q^2)\sigma_a^2$$

og autokovariansen er

$$\gamma_k = \begin{cases} (-\theta_k + \theta_1 \theta_{k+1} + \dots + \theta_{q-k} \theta_q) \sigma_a^2 & k = 1, 2, \dots, q \\ 0 & k > q \end{cases}$$

da  $E[a_t^2] = \sigma_a^2$  og  $E[a_t a_{t+k}] = 0$  for  $k \neq 0$ , fordi  $a_t$  og  $a_{t+k}$  er uafhængige for  $k \neq 0$ , da  $a$ 'erne er *tilfældige*. Det vil sige, at autokorrelationerne bliver

$$\rho_k = \begin{cases} \frac{-\theta_k + \theta_1 \theta_{k+1} + \dots + \theta_{q-k} \theta_q}{1 + \theta_1^2 + \dots + \theta_q^2} & k = 1, 2, \dots, q \\ 0 & k > q \end{cases} \quad (11)$$

Af dette ses altså, at for en MA( $q$ )-proces er autokorrelationerne nul, for  $k > q$ . Dette kan anvendes i modelidentifikationen.

Det er også muligt at estimere parametrene i en MA( $q$ )-proces ud fra autokorrelationerne. Imidlertid fremkommer der ikke et simpelt udtryk, da ligningerne, som det ses af Ligning (11), ikke er lineære - hvilket de var for AR( $p$ )-processen.

**Partielle autokorrelationer for en glidende midlings proces** Man kan vise, se Box and Jenkins (1970) kapitel 3, at den partielle autokorrelationsfunktion for en MA-proces er uendelig, at den er domineret af dæmpede eksponential- og/eller sinusfunktioner, samt at den langsomt konvergerer mod nul.

### Blandet autoregressiv-glidende midlings modeller

Det er ofte nødvendigt, at man både har en autoregressiv og en glidende midlings model bygget sammen i én. Denne model kan da skrives med de to operatorer

$$\phi(B)\tilde{z}_t = \theta(B)a_t \quad (12)$$

Skrives dette udtryk ud og isoleres  $\tilde{z}_t$ , fås

$$\tilde{z}_t = \phi_1 \tilde{z}_{t-1} + \dots + \phi_p \tilde{z}_{t-p} + a_t - \theta_1 a_{t-1} - \dots - \theta_q a_{t-q} \quad (13)$$

Dette er en model med  $p+q+2$  parametre, som igen må bestemmes ud fra data. Modellen kaldes for en *AutoRegressive Moving Average model af orden (p, q)* eller blot ARMA( $p, q$ ).

Disse modeller for  $\tilde{z}_t$  kan også skrives som en uendelig lineær sum af  $a_t$ 'er

$$\tilde{z}_t = a_t + \sum_{j=1}^{\infty} \psi_j a_{t-j} \quad (14)$$

da enhver autoregressiv model kan udtrykkes ved en uendelig sum af  $a_t$ 'er, se Afsnit 6.2. Alternativt kan modellerne også skrives som en uendelig sum af  $z_t$ 'er plus et chok,  $a_t$ ,

$$\tilde{z}_t = \sum_{j=1}^{\infty} \pi_j \tilde{z}_{t-j} + a_t \quad (15)$$

da det følger af Ligning (14), at ethvert  $a_t$  kan skrives som  $\tilde{z}_t$  plus en uendelig sum af tidligere  $a_t$ 'er.

**Sammenhængen mellem  $\psi$  og  $\pi$  vægtene** For at finde sammenhængen mellem de to slags vægte indføres *backward shift* operatoren,  $B$ , i Ligning (14), derved fås

$$\tilde{z}_t = (1 + \sum_{j=1}^{\infty} \psi_j B^j) a_t$$

hvilket også kan skrives

$$\tilde{z}_t = \psi(B) a_t \quad (16)$$

hvor

$$\psi(B) = (1 + \sum_{j=1}^{\infty} \psi_j B^j)$$

På samme måde kan Ligning (15) skrives

$$a_t = (1 - \sum_{j=1}^{\infty} \pi_j B^j) \tilde{z}_t$$

eller

$$a_t = \pi(B) \tilde{z}_t \quad (17)$$

hvor

$$\pi(B) = (1 - \sum_{j=1}^{\infty} \pi_j B^j)$$

Opererer vi nu med  $\psi(B)$  på begge sider af Ligning (17), fås

$$\psi(B) a_t = \psi(B) \pi(B) \tilde{z}_t$$

hvilket ifølge Ligning (16) er lig

$$\tilde{z}_t = \psi(B) \pi(B) \tilde{z}_t$$

af dette følger at

$$\psi(B) \pi(B) = 1$$

eller

$$\pi(B) = \psi^{-1}(B)$$

Hermed har vi fundet sammenhængen mellem  $\pi$ - og  $\psi$ -vægtene.

**Generelle udtryk for henholdsvis  $\psi$ - og  $\pi$ -vægtene** Opereres på begge sider af Ligning (16) med  $\phi$  fås

$$\phi(B) \tilde{z}_t = \phi(B) \psi(B) a_t$$

da

$$\phi(B) \tilde{z}_t = \theta(B) a_t$$

følger at

$$\theta(B) = \phi(B) \psi(B)$$

der også kan skrives

$$(1 - \theta_1 B - \dots - \theta_q B^q) = (1 - \phi_1 B - \dots - \phi_p B^p)(1 + \psi_1 B + \psi_2 B^2 + \dots)$$

Dette medfører at koefficienterne i  $\psi(B)$  fås ved at sætte koefficienterne for  $B$  på højre og venstre side af lighedstegnet lig hinanden.

Ganges med  $\theta$  i Ligning (17) fås analogt med ovenfor

$$(1 - \phi_1 B - \dots - \phi_q B^q) = (1 - \theta_1 B - \dots - \theta_p B^p)(1 - \pi_1 B - \pi_2 B^2 - \dots)$$

Igen findes koefficienterne for  $\pi$  ved at sammenstille koefficienterne for  $B$  på højre og venstre side af lighedstegnet.

**Autokorrelationerne for en blandet proces** Følges den samme metode som i Afsnit 6.2, fås, ved at gange igennem med  $\tilde{z}_{t-k}$  i Ligning (13), følgende udtryk for autokovariansfunktionen

$$\gamma_k = \phi_1 \gamma_{k-1} + \dots + \phi_p \gamma_{k-p} + \gamma_{za}(k) - \theta_1 \gamma_{za}(k-1) - \dots - \theta_q \gamma_{za}(k-q) \quad (18)$$

hvor  $\gamma_{za}(k) = E[\tilde{z}_{t-k} a_t]$  er krydskovariansen mellem  $z$  og  $a$ . Da  $z_{t-k}$  kun er afhængig af chok, der er indtruffet, det vil sige chok op til tiden  $t-k$ , følger det, at

$$\begin{aligned} \gamma_{za}(k) &= 0 & k > 0 \\ \gamma_{za}(k) &\neq 0 & k \leq 0 \end{aligned}$$

Dette implicerer, at

$$\gamma_k = \phi_1 \gamma_{k-1} + \dots + \phi_p \gamma_{k-p}, \quad k \geq q+1$$

og autokorrelationerne kan således skrives

$$\rho_k = \phi_1 \rho_{k-1} + \dots + \phi_p \rho_{k-p}, \quad k \geq q+1 \quad (19)$$

eller

$$\phi(B)\rho_k = 0, \quad k > q$$

Af dette følger, fra Ligning (18), at for en ARMA( $p, q$ )-proces er der  $q$  autokorrelationer,  $\rho_1, \dots, \rho_q$ , hvis værdier er direkte afhængige af de  $p$  parametre, der indgår i AR-modellen, men også af de  $q$  parametre i MA-modellen. De  $p$  værdier,  $\rho_q, \dots, \rho_{q-p+1}$ , giver de nødvendige startbetingelser til differensligningen (19), der for  $k \geq q+1$  helt bestemmer værdierne af autokorrelationerne. Er  $q-p < 0$ , vil autokorrelationerne,  $\rho_j, j = 0, 1, 2, \dots$  bestå af en blanding af eksponentielt dæmpede funktioner og/eller dæmpede sinusfunktioner (se Afsnit 6.2), hvis natur bestemmes af de  $p$  parametre  $\phi_1, \dots, \phi_p$  i ARMA( $p, q$ )-processen. Er derimod  $q-p \geq 0$ , vil der være  $q-p-1$  autokorrelationer,  $\rho_0, \rho_1, \dots, \rho_{q-p}$ , der ikke følger dette mønster. Alle disse fakta er vigtige i identifikationen af modellen.

**Den partielle autokorrelationsfunktion for en blandet proces** Processen i Ligning (12) kan skrives

$$a_t = \theta^{-1}(B)\phi(B)\tilde{z}_t$$

Da  $\theta^{-1}(B)$  er en uendelig serie i  $B$ , er den partielle autokorrelationsfunktion også uendelig. Den opfører sig som den partielle autokorrelationsfunktion for en ren MA-proces, der er domineret af en blanding af eksponentielt dæmpede funktioner og/eller dæmpede sinusfunktioner, afhængigt af de i modellen indgående parametre.

### Invertibilitet og stationaritet

Som det er vist i Afsnit 6.2, kan enhver AR-proces skrives som en *uendelig* sum af  $a_t$ 'er, men også som en endelig sum af  $z_t$ 'er og enhver MA-proces kan skrives som en *uendelig* sum af  $z_t$ 'er, foruden som en endelig sum af  $a_t$ 'er. Det, at disse uendelige summer skal konvergere, medfører nogle begrænsninger af parametrene i de to processer. For MA-processer medfører dette *invertibilitets* betingelsen og for AR-processer *stationaritets* betingelsen.

**Invertibilitet** Invertibilitetsbetingelsen for en MA-proces udledes på følgende måde: Da en MA-proces skrives

$$\theta(B)a_t = \tilde{z}_t$$

fås ved *invertering* af denne ligning, at

$$a_t = \theta^{-1}(B)\tilde{z}_t$$

Det vil sige, at hvis

$$\theta(B) = \prod_{j=1}^q (1 - H_j B)$$

så er<sup>18</sup>

$$\pi(B) \equiv \theta^{-1}(B) = \sum_{j=1}^q \frac{M_j}{1 - H_j B} = \sum_{j=1}^q \left( M_j \sum_{i=1}^{\infty} H_j^i B^i \right)$$

der er konvergent, hvis  $|H_j| < 1$ , for  $j = 1, 2, \dots, q$ . Da rødderne til  $\theta(B) = 0$  er  $H_j^{-1}$ , betyder invertibilitets betingelsen for en MA-proces, at rødderne i  $\theta(B) = 0$  skal ligge *udenfor* enhedscirklen. Da AR-processen er udtrykt ved en endelig sum af  $\tilde{z}_t$ 'er, er *enhver* AR-proces invertibel.

**Stationaritet** Er en tidsserie stationær, kan man vise, at autokovarians- og autokorrelationsmatricerne begge er positivt definitte<sup>19</sup>. Man kan også vise, se Box and Jenkins (1970) appendix A3.1 og Grenander and Rosenblatt (1957), at hvis serien  $\psi(B)$  konvergerer for  $|B| \leq 1$ , så er de ovennævnte betingelser opfyldt.

For en *autoregressiv model af orden p*

$$(1 - \phi_1 B - \dots - \phi_p B^p)\tilde{z}_t = \phi(B)\tilde{z}_t = a_t \quad (20)$$

kan  $\phi(B)$  omskrives til

$$\phi(B) = (1 - G_1 B)(1 - G_2 B) \dots (1 - G_p B)$$

hvor  $G_i$  er den inverse til den  $i$ 'te rod i ligningen  $\phi(B) = 0$ . Dette betyder at Ligning (20) kan skrives

$$\tilde{z}_t = \phi^{-1}(B)a_t = \sum_{i=1}^p \frac{K_i}{(1 - G_i)} a_t = \sum_{i=1}^p \left( K_i \sum_{j=0}^{\infty} G_i^j B^j \right) a_t$$

hvilket medfører, at skal serien  $\psi(B) = \phi^{-1}(B)$  være konvergent for  $|B| \leq 1$ , fås at  $|G_i| < 1$ . Da  $G_i$ 'erne er de inverse til rødderne i den *karakteristiske ligning*,  $\phi(B) = 0$ , gælder der om rødderne, at de skal ligge *udenfor* enhedscirklen. Da en MA-proces er udtrykt ved en *endelig* sum af  $a_t$ 'er, er *enhver* MA-proces stationær.

**Sammenfatning** For at få overblik over hvordan autokorrelationerne og de partielle autokorrelationer opfører sig for en given model, bringes her en sammenfatning.

For *autoregressive*-processer er

- *autokorrelationsfunktionen* uendelig. Den består af dæmpede eksponentialfunktioner og/eller dæmpede sinusbølger.
- *den partielle autokorrelationsfunktion* endelig. Fra  $k > p$  er funktionen 0.

For *glidende midlings*-processer er

<sup>18</sup>  $M_j$  bestemmes ved partiel opsplitning (eng. partial fractioning). Den sidste lighed, fremkommer ved direkte division.

<sup>19</sup> Om definitionen af autokovarians- og autokorrelationsmatricerne og beviset, se Afsnit 6.



- *autokorrelationsfunktionen* endelig. Fra  $k > q$  er funktionen 0.
- *den partielle autokorrelationsfunktion* uendelig. Den er domineret af dæmpede eksponentialfunktioner og/eller dæmpede sinusbølger.

For *blandede autoregressive og glidende midlings*-processer er

- *autokorrelationsfunktionen* uendelig. Den består af dæmpede eksponentialfunktioner og/eller dæmpede sinusbølger for  $k > q - p$ .
- *den partielle autokorrelationsfunktion* uendelig. Den er domineret af dæmpede eksponentialfunktioner og/eller dæmpede sinusbølger for  $k > q - p$ .

### 6.3 Ikke-stationære tidsserier

Oftentimes kan de tidsserier, man møder i "det virkelige liv", ikke betragtes som varierende om en fast middelværdi. Det vil sige tidsserierne er *ikke* stationære, hvilket indtil nu har været et kardinalpunkt i udledningen. Det vil imidlertid vise sig, at man, ved at slække på nogle af kravene, også kan lave stokastiske modeller for ikke-stationære tidsserier. Tillader man rødderne i den karakteristiske ligning,  $\phi(B) = 0$ , at ligge indenfor enhedscirklen, vokser modellen efter et kort tidsrum eksponentielt. Det er ikke tidsserier af denne art, vi ønsker at modellere. Der er nu kun den mulighed tilbage, at rødderne ligger *på* enhedscirklen, da det blev vist i sidste afsnit fandt, at rødderne for en stationær proces ligger *udenfor* enhedscirklen.

Betrægter man modellen

$$\varphi(B)\tilde{z}_t = \theta(B)a_t$$

hvor  $\varphi(B)$  er en ikke-stationær autoregressiv operator, der har  $d$  af sine rødder på enhedscirklen og resten udenfor, kan den skrives

$$\varphi(B)\tilde{z}_t = \phi(B)(1-B)^d\tilde{z}_t = \theta(B)a_t$$

hvor  $\phi(B)$  er en *stationær* autoregressiv operator. Da  $\tilde{z}_t = z_t - \mu$ , hvor  $\mu$  er middelværdien, gælder der, at  $\nabla^d\tilde{z}_t = \nabla^dz_t$ , for  $d \geq 1$ .  $\nabla$  kaldes *baglæns differens operatoren* og er defineret ved

$$\nabla z_t = z_t - z_{t-1} = (1-B)z_t$$

Dette medfører, at modellen kan skrives

$$\phi(B)\nabla^dz_t = \theta(B)a_t \quad (21)$$

eller

$$\phi(B)w_t = \theta(B)a_t$$

hvor

$$w_t = \nabla^dz_t \quad (22)$$

Resultatet er, at man kan forsøge at gøre en ikke-stationær tidsserie stationær ved at anvende operatoren  $\nabla$ . Ligning (21) beskriver den model, der kaldes den *AutoRegressive Integrated Moving Average process* af orden  $(p, d, q)$ , eller kort *ARIMA* $(p, d, q)$ -processen. Skriver man udtrykket ud, får man

$$\nabla^dz_t = w_t = \phi_1w_{t-1} + \dots + \phi_pw_{t-p} + a_t - \theta_1a_{t-1} - \dots - \theta_qa_{t-q} \quad (23)$$

ARIMA-modellen for ikke-stationære tidsserier er af fundamental vigtighed i problemer, der vedrører forudsigelser af tidsserier og kontrol af industrielle processer, se for eksempel Yaglom (1955), Box og Jenkins (1962), Box og Jenkins (1963), Box et al. (1967), Box og Jenkins (1968a), Box og Jenkins (1968b), Box og Jenkins (1969). Bemærk, at ARIMA-processen som specialtilfælde indeholder både den



autoregressive proces og glidende midlings processen, og at den også indeholder både stationære og ikke-stationære modeller.

Grunden til I'et (Integrated) i forkortelsen er følgende: Tager man den inverse af Ligning (22) fås

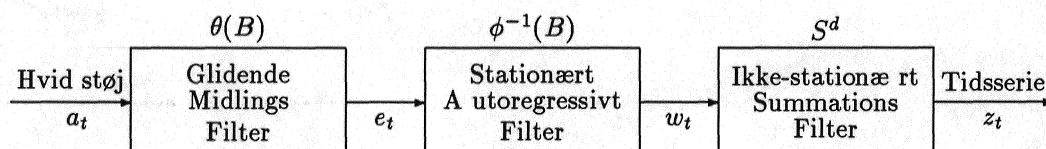
$$z_t = (\nabla^d)^{-1} w_t = S^d w_t$$

hvor

$$S w_t = \sum_{j=0}^{\infty} w_{t-j} = w_t + w_{t-1} + \dots$$

er den inverse til  $\nabla$ . Som det ses, er dette en sum (integral), heraf I'et.

ARIMA-processen er illustreret skematisk i Figur 25.



Figur 25. Skematisk fremstilling af ARIMA-processen, hvor de enkelte operatoren betragtes som filtre. Efter idé af Box og Jenkins, 1970

Den kan betragtes som bestående af tre filtre

1. Et filter med input  $a_t$ , transfer funktion  $\theta(B)$  og output  $e_t = \theta(B)a_t$
2. Et filter med input  $e_t$ , transfer funktion  $\phi^{-1}(B)$  og output  $w_t = \phi^{-1}e_t$
3. Til sidst et filter med input  $w_t$ , transfer funktion  $S^d$  og output  $z_t$  giver ved Ligning (23).

## 6.4 Modeller for tidsserier med periodiske svingninger

Da det vides om den tidsserie, der skal modelleres i denne rapport, at den indeholder en periodisk svingning, se Afsnit 6.1, er det nødvendigt at inkludere periodicitet i modellen.

Det, der karakteriserer en tidsserie med periode  $s$ , er, at observationer, der er  $s$  intervaller fra hinanden, er ens. Derfor vil man forvente, at operatoren,  $B^s z_t = z_{t-s}$ , er vigtig. Da det endvidere er sandsynligt, at tidsserien ikke er stationær, vil operatoren  $\nabla_s z_t = (1 - B^s)z_t = z_t - z_{t-s}$  også forventes at spille en rolle. Operatoren  $1 - B^s$  har  $s$  nulpunkter  $e^{i(2\pi k/s)}$ ,  $k = 0, 1, \dots, s-1$ .

Den ARIMA-model, man får frem ved at bruge de førnævnte operatoren, vil have følgende udseende

$$\Phi(B^s) \nabla_s^D z_t = \Theta(B^s) \alpha_t \quad (24)$$

Hvorved observationer med  $s$  intervaller imellem bliver relateret til hinanden.  $\Phi(B^s)$  og  $\Theta(B^s)$  er polynomier i  $B^s$  henholdsvis af orden  $P$  og  $Q$ . For temperaturserien er  $s = 365$ . Samtidigt vil det forventes, at  $z_{t-1}$  og  $z_{t-1-s}$  er ens. Hvilket

giver anledning til modellen

$$\Phi(B^s)\nabla_s^D z_{t-1} = \Theta(B^s)\alpha_{t-1} \quad (25)$$

og så videre for alle  $s$ . Imidlertid er det meget sandsynligt, at operatorerne  $\Phi$  og  $\Theta$  for hver af disse modeller er ens.

Fejlene i den resulterende model,  $\alpha_t, \alpha_{t-1}, \dots$ , kan ikke forventes at være ukorrelerede, idet  $z_t$  må forventes at være relateret til både  $z_{t-1}$  og  $z_{t-2}$  og så videre, det vil sige, at  $\alpha_t$  i Ligning (24) er relateret til  $\alpha_{t-1}$  i Ligning (25), og så videre for resten af de  $s$  ligninger. På grund af dette indføres nu en ny model

$$\phi(B)\nabla^d \alpha_t = \theta(B)a_t$$

hvor  $a_t$ 'erne er hvid støj, og  $\phi(B)$  og  $\theta(B)$  er polynomier i  $B$  henholdsvis af orden  $p$  og  $q$ . Indsættes dette i Ligning (24), fås

$$\phi(B)\Phi(B^s)\nabla^d \nabla_s^D z_t = \theta(B)\Theta(B^s)a_t$$

Denne model kaldes en *multiplikativ ARIMA-proces af orden  $(p, d, q) \times (P, D, Q)_s$* . Det er denne model, der vil blive anvendt i det følgende. Er der flere periodiske komponenter i serien, kan ovenstående argumentation generaliseres til også at gælde disse.

## 7 Modeludvikling

Udviklingen af modellen for temperaturserien foregår i 3 faser:

1. Modelidentifikation
2. Estimering af parametrene i modellen
3. Diagnostisk undersøgelse af modellen

Om dette handler de 3 næste afsnit. Meget af den teori, der blev udviklet i det foregående afsnit, vil blive anvendt. Det er imidlertid nødvendigt at generalisere en del af den, da den model, der skal bestemmes hér, er en multiplikativ model.

### 7.1 Modelidentifikation

I den fase af modeludviklingen, der kaldes modelidentifikationen, udvikles en første tentativ model. Dette gøres ved at studere plot af autokorrelationsfunktionen og den partielle autokorrelationsfunktion for de tidsserier, der fremkommer, efter at forskellige differensoperatorer har virket på temperaturserien.

**Anvendelse af autokorrelationsfunktionen og den partielle autokorrelationsfunktion til modelidentifikation** Som det blev vist i Afsnit 6.2, gælder der for en stationær blandet autoregressiv-glidende midlings-proces af orden  $(p, 0, q)$ ,  $\phi(B)\tilde{z}_t = \theta(B)a_t$ , at autokorrelationsfunktionen opfylder følgende ligning

$$\phi(B)\rho_k = 0, \quad k > q$$

Det vil sige, at hvis  $\phi(B) = \prod_{i=1}^p (1 - G_i B)$ , så kan løsningen til denne differensligning skrives

$$\rho_k = A_1 G_1^k + A_2 G_2^k + \dots + A_p G_p^k, \quad k > q - p \quad (26)$$

Stationaritetsbetingelsen kræver, at  $\phi(B)$ 's rødder skal ligge udenfor enhedscirklen, hvilket medfører, at  $G_i$ 'erne skal ligge indenfor enhedscirklen.

Ligning (26) viser, at hvis ingen af  $G_i$ 'erne ligger tæt ved yderkanten af enhedscirklen, vil  $\rho_k$  hurtigt gå mod 0, når  $k$  vokser. Er derimod for eksempel  $G_1$  meget tæt på 1, således at

$$G_1 = 1 - \delta$$

hvor  $\delta$  er lille og positiv<sup>20</sup>, vil

$$\rho_k \simeq A_1(1 - k\delta)$$

for store  $k$ . Dette betyder, at autokorrelationsfunktionen ikke hurtigt går mod 0, men derimod aftager lineært. Og tilsvarende hvis der er flere rødder, der er tæt på 1.

Det vil sige, at har autokorrelationerne en tendens til ikke hurtigt at ville dø ud, så tyder det på, at der er en rod tæt på 1. I Afsnit 6.3 blev det vist, at dette netop betød, at tidsserien ikke var stationær. Der er meget, der taler for at anvendes en passende differensoperator, så er den resulterende tidsserie stationær. På dette område er der ikke nogen væsentlig forskel på, om modellen er multiplikativ eller ej, argumentationen er den samme.

Vi skal altså forsøge at gøre temperaturserien stationær. Dette gøres ved at studere autokorrelationerne, for henholdsvis temperaturserien og den resulterende tidsserie, efter at operatorene  $\nabla$ ,  $\nabla^2$ ,  $\nabla_{365}$ ,  $\nabla_{365}^2$ ,  $\nabla\nabla_{365}$ ,  $\nabla^2\nabla_{365}$ ,  $\nabla\nabla_{365}^2$ ,  $\nabla^2\nabla_{365}^2$  er anvendt. Som det blev vist i Afsnit 5 afhænger udseendet af autokorrelationsfunktionen af hvilken model, der anvendes på den resulterende tidsserie. Det er altså nødvendigt at tage dette i betragtning, når temperaturserien forsøges gjort stationær.

For temperaturserien ses det klart, at autokorrelationsfunktionen ikke dør ud hurtigt, se Figur 26, det vil sige temperaturserien er ikke stationær; endvidere ses det, at funktionen er periodisk med periode 365, hvilket næppe kan virke overraskende.

Prøves med  $\nabla z_t$  alene, får den resulterende tidsserie en autokorrelationsfunktion som vist i Figur 27. På denne ses det, at det nu er lykkedes at "slå" de fleste af autokorrelationerne ned, for  $k > 1$ . Bartlett's formel, se Ligning (6), for eksempel en MA(1)-model, giver, at  $2\sigma = 0.011$ . Det vil sige at autokorrelationer, der er numerisk mindre end dette, kan betragtes som værende nul. Endnu ses det, at der, for  $k = 365p$ ,  $p = 1, 2, \dots$ , stadig er autokorrelationer, der er signifikant forskellige fra nul.

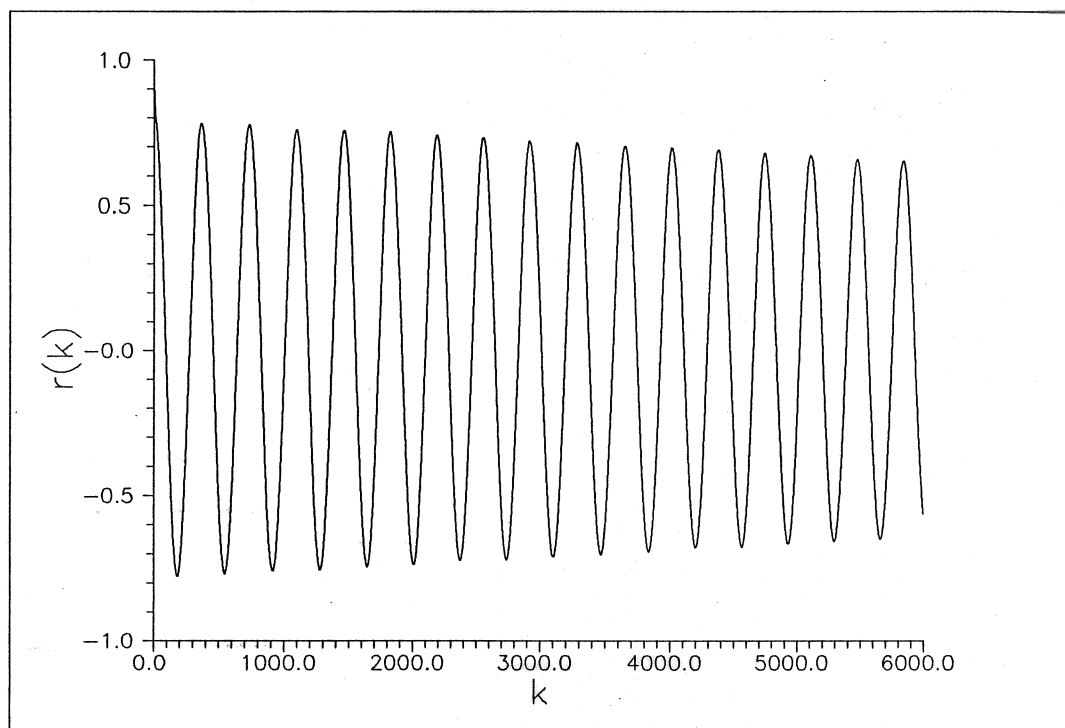
Anvendes operatoren,  $\nabla^2$ , fås, at der også er for mange autokorrelationer, der er større end nul, ligegyldigt hvilken model der anvendes.

Introduceres nu  $\nabla_{365}$  fås autokorrelationerne vist i Figur 28. Som denne viser, er der også her (for) mange autokorrelationer, der er større end nul, ligegyldigt modellen. Tilsvarende for  $\nabla_{365}^2$ .

Da anvendelse af hverken  $\nabla$  eller  $\nabla_{365}$  hver for sig kan gøre temperaturserien stationær, er det nærliggende at forsøge med en kombination af disse. Efter at have anvendt operatoren  $\nabla_{365}\nabla$ , fås autokorrelationerne vist i Figur 29.

Det er tydeligt, at ikke alle koefficienterne er nul. Dette gør imidlertid ikke noget, for anvender man for eksempel en MA(1)×MA(1)<sub>365</sub>-model, må  $r_0, r_1, r_{364}, r_{365}, r_{366}$  være forskellige fra nul. Bartlett's formel giver  $2\sigma[r_k] = 2.06 \cdot 10^{-2}$ , sammenlignes dette med værdien af  $r_{367} = 4.8 \cdot 10^{-2}$ , ses det, at denne ikke kan betragtes som værende forskellig fra nul. Det samme gælder for  $r_2 = -9.3 \cdot 10^{-2}$  og  $r_{363} = 4.2 \cdot 10^{-2}$ , selvom Ligning (6) ikke gælder her. Det vil sige den foreslåede model må forkastes. Prøves med MA(2)×MA(1)<sub>365</sub>, må følgende autokorrelationer være forskellige fra nul:  $r_0, r_1, r_2, r_{363}, r_{364}, r_{365}, r_{366}, r_{367}$ , spredningen er  $2\sigma[r_k] = 2.07 \cdot 10^{-2}$ , for  $k > 367$ , heller ikke for denne model er betingelserne opfyldt, idet  $r_{368} = 3.0 \cdot 10^{-2}$ , endvidere er  $r_3 = -5.7 \cdot 10^{-2}$  og

<sup>20</sup>Så lille, at også  $k\delta$  er lille, for  $k$  stor.



Figur 26. Autokorrelationsfunktionen for temperaturserien  $z_t$ .

$r_{362} = 3.4 \cdot 10^{-2}$ . Forsøges med  $MA(3) \times MA(1)_{365}$  må følgende autokorrelationer være forskellige fra nul:  $r_0, r_1, r_2, r_3, r_{361}, r_{362}, r_{363}, r_{364}, r_{365}, r_{366}, r_{367}, r_{368}$ . Her viser det sig, at betingelserne er opfyldt; der er imidlertid det problem, at ifølge Box and Jenkins (1970) er det sjældent, at man behøver modeller, hvor  $q > 2$ . Samtidigt er et vigtigt princip i modeludviklingen, at man er sparsommelig med antallet af parametre i modellen. Det vil sige man skal vælge en model med så få parametre som muligt.

For ikke at bryde med dette princip kunne man få den ide at prøve at bruge en differensoperator af højere orden. Forsøges med  $\nabla^2 \nabla_{365}$ , fås autokorrelationerne vist i Figur 30. Udseendet af grafen minder meget om den tilsvarende graf for  $\nabla \nabla_{365}$ , forskellen er, at værdierne omkring 0 og 365 aftager hurtigere, når operatoren  $\nabla^2 \nabla_{365}$  anvendes. For  $MA(1) \times MA(1)_{365}$  fås  $2\sigma[r_k] = 2.96 \cdot 10^{-2}$  og  $r_0, r_1, r_{364}, r_{365}, r_{366}$  må - som før - være forskellige fra nul. Det viser sig at  $r_2$  også denne gang må betragtes som værende forskellig fra 0, hvilket igen fører til at modellen må forkastes. Forsøges nu med  $MA(2) \times MA(1)_{365}$ , fås  $2\sigma = 2.2 \cdot 10^{-2}$ . Nu må  $r_0, r_1, r_2, r_{363}, r_{364}, r_{365}, r_{366}, r_{367}$  være forskellige fra 0, det er de med undtagelse af  $r_{363}$  og  $r_{367}$ , der er mindre end  $2\sigma$ , dette kunne betyde, at  $MA(1) \times MA(1)_{365}$  også var en model, der kunne bruges. Nu forsøges imidlertid med  $MA(2) \times MA(1)_{365}$ .

Studerer man de partielle autokorrelationer ses det, at disse langsomt klinger af, se Figur 31. På figuren ses også et udslag for  $k=365$ , hvilket var at forvente. Dette underbygger yderligere, at den søgte model er en MA-model, da det gælder for disse, at den partielle autokorrelationsfunktion er uendelig og domineret af dæmpede eksponentialfunktioner og/eller dæmpede sinusbølger, se side 47.

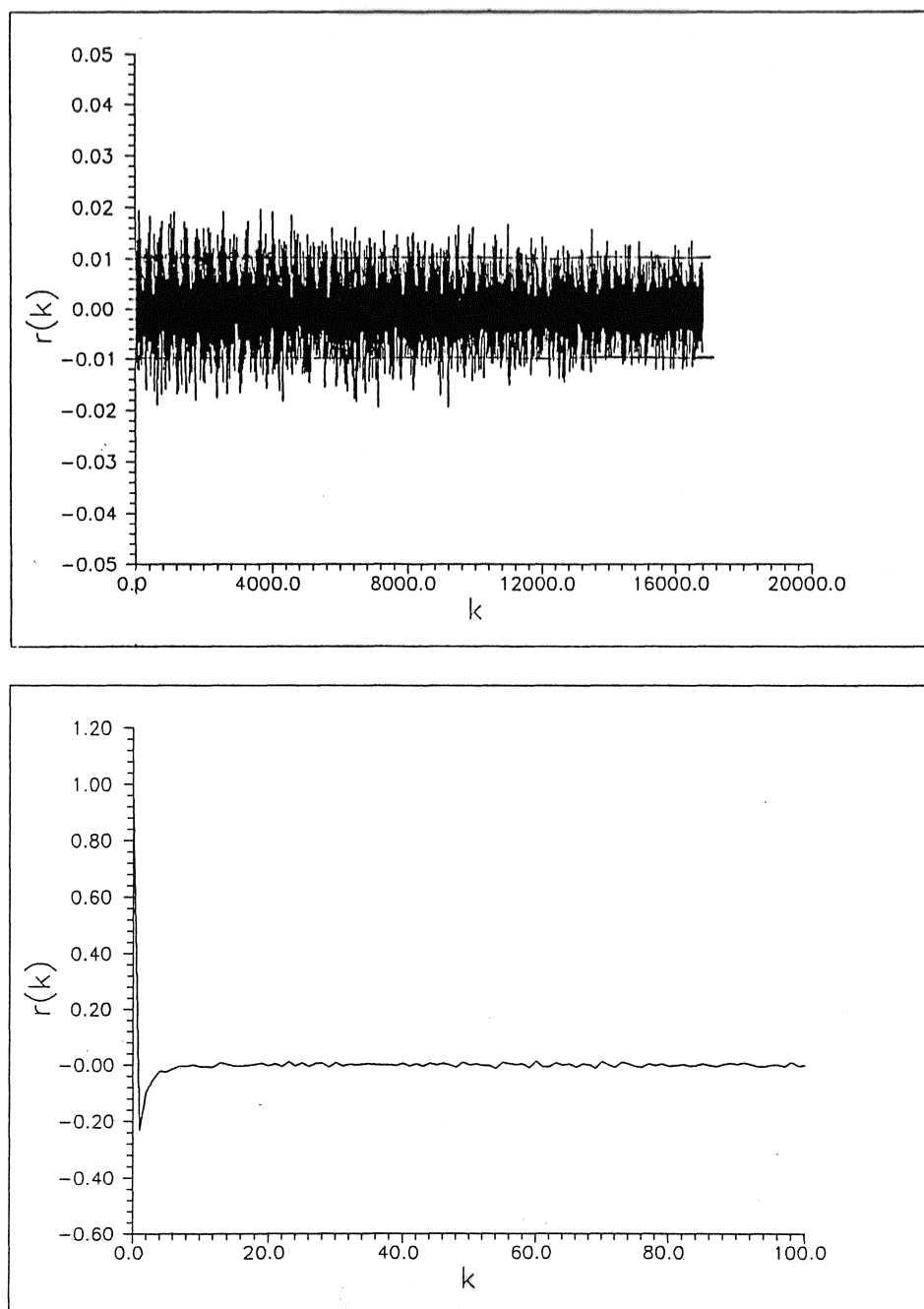
Konklusionen er altså, at vi ved at studere autokorrelationerne, er kommet frem til følgende model

$$w_t = (1 - \theta_1 B - \theta_2 B^2)(1 - \Theta B^{365})a_t$$

hvor

$$w_t = \nabla^2 \nabla_{365} z_t$$





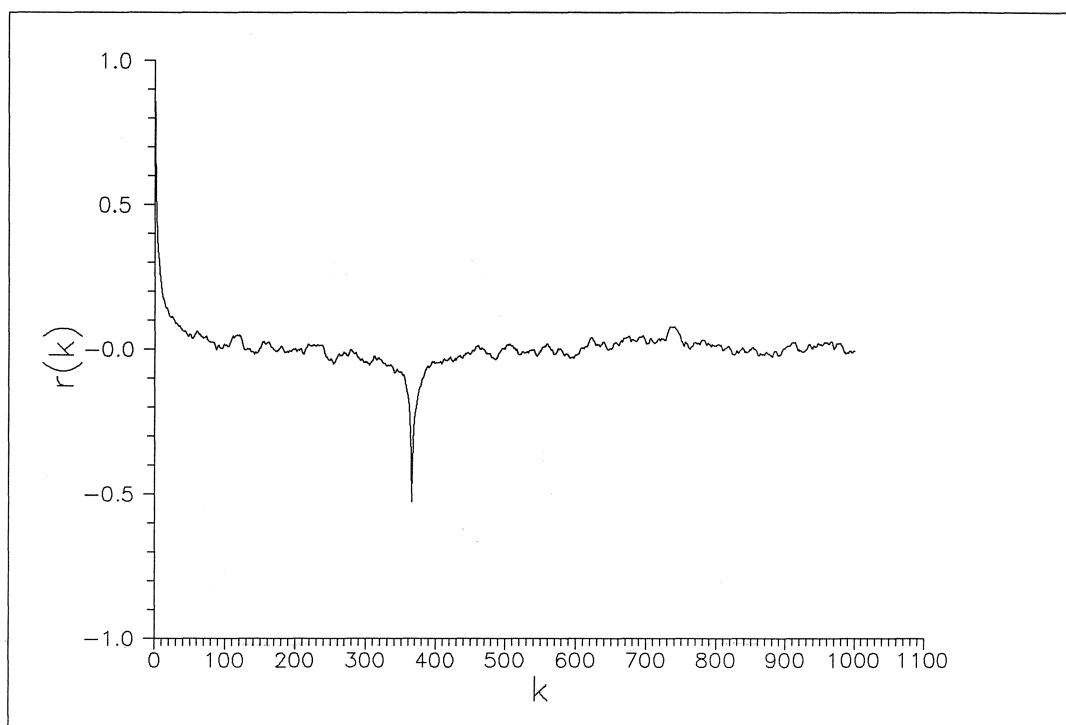
Figur 27. Autokorrelationsfunktionen for tidsserien  $\nabla z_t$ , øverst. Nederst er vist et udsnit for  $k = 0, \dots, 100$  af den samme funktion.

## 7.2 Estimering af parametrene

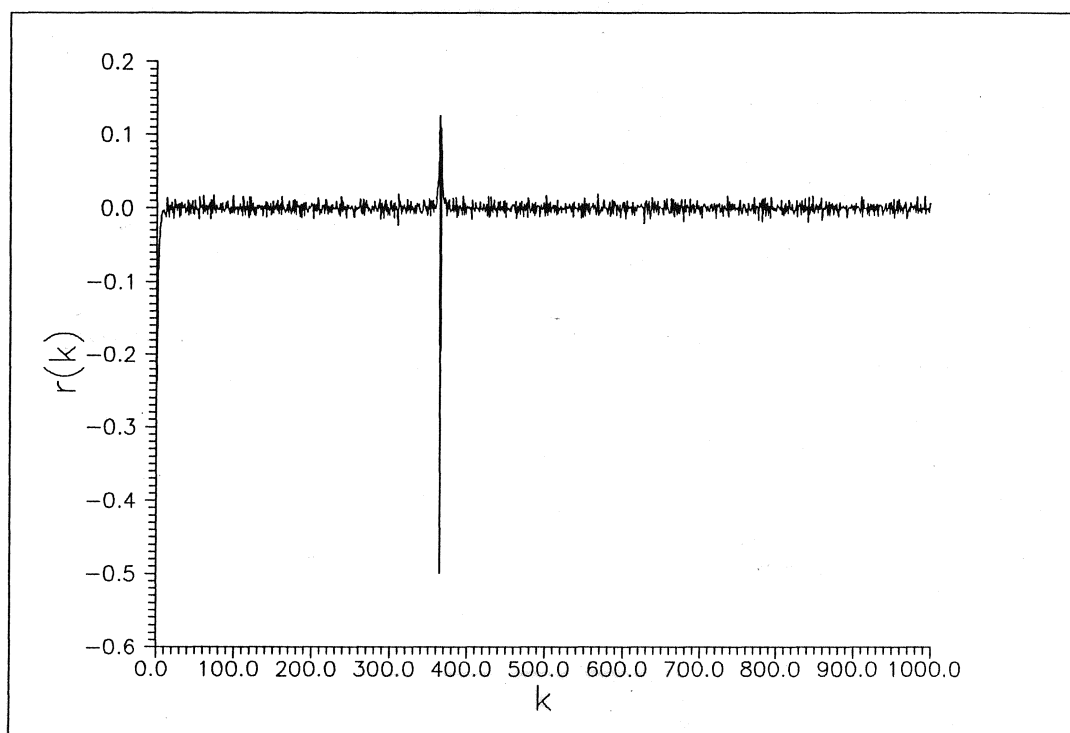
Næste skridt i modeludviklingen er at finde værdien af parametrene. Parametrene bestemmes ud fra *likelihood princippet*, se Box and Jenkins (1970) side 209 og Fisher (1956), Barnard (1949), Birnbaum (1962). Dette princip siger, hvis den fundne model er korrekt, at alt, *data*'ene siger om parametrene, er indeholdt i likelihood funktionen, og at de parametre, der maksimere denne, er dem, der giver den mindst fejlagtige model.

Det er nødvendigt at undersøge likelihood funktionen grundigt for at undgå, at det maksimum, der findes, ikke er et lokalt maximum. Typisk vil man studere funktionen grafisk - dette er imidlertid lidt svært, da funktionen, for den hér valgte model, er 4 dimensional! Kigger man på tabulerede værdier af likelihood





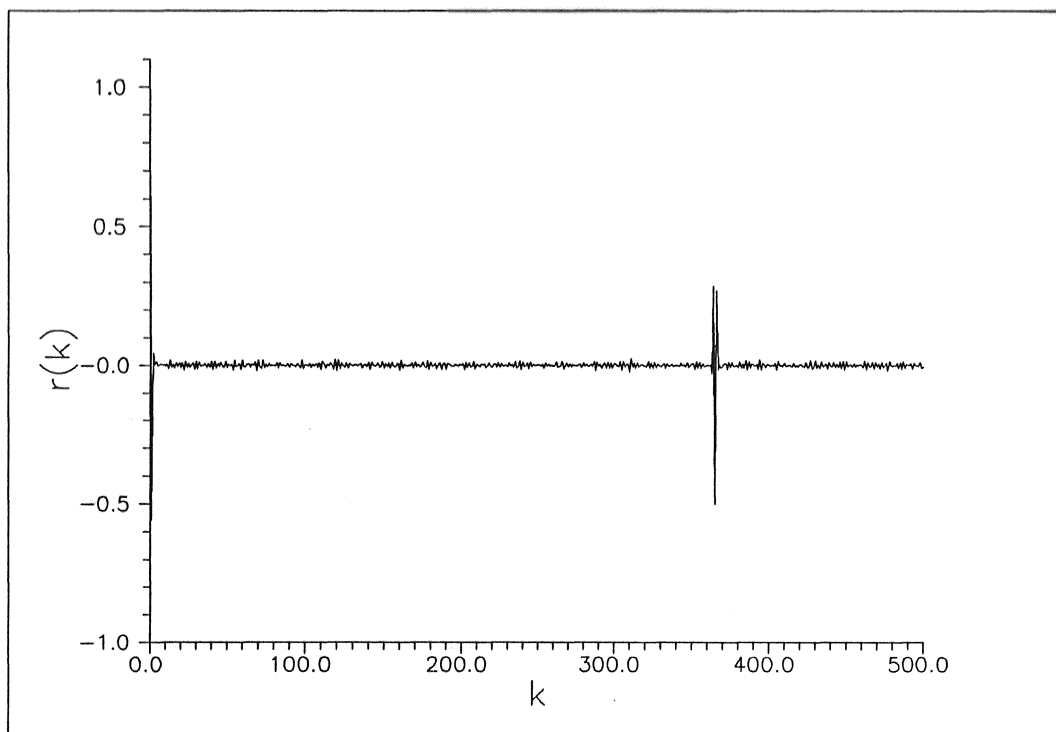
Figur 28. Autokorrelationsfunktionen for tidsserien  $\nabla_{365} z_t$ .



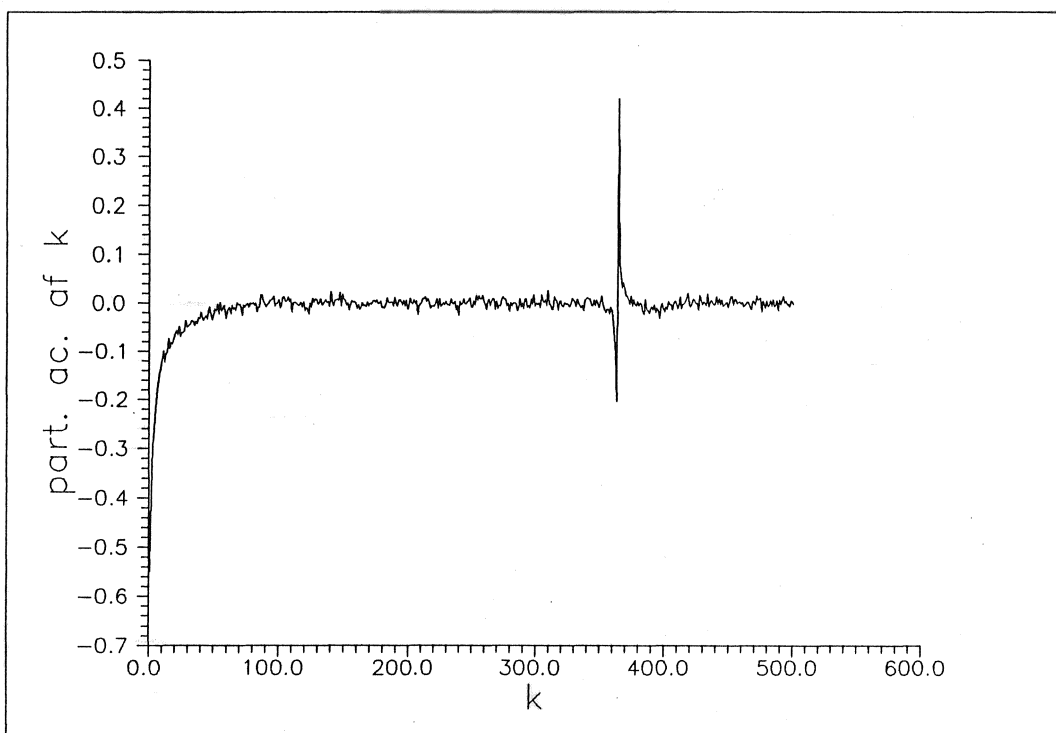
Figur 29. Autokorrelationsfunktionen for tidsserien  $\nabla \nabla_{365} z_t$ .

funktionen, får man dog det indtryk, at der er ét globalt maximum og ingen lokale maxima.

En mere objektiv måde er at maksimere likelihood funktionen ved en numerisk metode, der initialiseres med flere forskellige startgæt. Giver dette det samme



Figur 30. Autokorrelationsfunktionen for tidsserien  $\nabla^2 \nabla_{365} z_t$ .



Figur 31. Den partielle autokorrelationsfunktion for tidsserien  $\nabla^2 \nabla_{365} z_t$ .

resultat, er det sandsynligt, at det fundne resultat maksimerer likelihood funktionen globalt. Især når det sammenholdes med det faktum, at likelihood funktioner, for store datamængder, ofte kun har ét maximum, se Box and Jenkins (1970) side 209. De fundne parametre kaldes *maximum likelihood (ML) estimerterne*. Den

numeriske metode, der anvendes til at maksimere likelihood funktionen, er igen "steepest descent" metoden, da det viser sig, at maksimeringsproblemet bliver et minimeringsproblem.

### Den ubetingede likelihood funktion for en ARIMA-proces

Antag, at vi har  $N = n + d + Ds$  observationer, der tilsammen danner tidsserien<sup>21</sup>  $\mathbf{z}$  og skrives  $z_{-d-Ds+1}, \dots, z_0, z_1, \dots, z_n$ . Det antages, at denne tidsserie er genereret af en ARIMA-proces af orden  $(p, d, q) \times (P, D, Q)_s$ , hvor  $s$  er perioden. Det er muligt ud fra disse observationer at danne  $\mathbf{w}$ , bestående af  $n = N - d - Ds$  differencer  $w_1, \dots, w_n$ , hvor  $w_t = \nabla^d \nabla_s^D z_t$ . Det vil sige vi skal bestemme parametrene  $\phi$  og  $\theta$  for den stationære ARMA( $p, q$ ) $\times(P, Q)_s$  model, der kan skrives

$$\phi_p(B)\Phi_P(B^s)\tilde{w}_t = \theta_q(B)\Theta_Q(B^s)a_t \quad (27)$$

hvor operatorernes index angiver deres orden. Isoleres  $a_t$  i dette udtryk fås:

$$a_t = \phi(B)\Phi(B^s)\tilde{w}_t + (T(B^s)(1 - t(B)) + t(B))a_t$$

hvor

$$T(B^s) = \Theta_1 B^s + \Theta_2 B^{2s} + \dots + \Theta_Q B^{Qs}$$

og

$$t(B) = \theta_1 B + \theta_2 B^2 + \dots + \theta_q B^q$$

og hvor  $w_t = \nabla^d \nabla_s^D z_t$ ,  $\tilde{w}_t = w_t - \mu$  og  $\mu = E[w_t]$ , der ofte er 0, når  $d > 0$ .

Problemet ved ovenstående er, at  $a_t$ 'erne ikke er entydigt bestemte, da værdien af dem afhænger af de startbetingelser, der anvendes i differensligningen. Det er imidlertid muligt at beregne  $a_t$ 'erne, når først de  $p + P$  værdier af  $w$ 'erne,  $\mathbf{w}_*$ , og de  $Qs + q$  værdier af  $a$ 'erne,  $\mathbf{a}_*$ , er givet. Man siger så, at  $a_t$  er beregnet *under betingelse af*  $\mathbf{w}_*$  og  $\mathbf{a}_*$ , dette skrives  $a_t(\phi, \theta | \mathbf{w}_*, \mathbf{a}_*, \mathbf{w})$ .

Antager man, at  $a_t$ 'erne er normalfordelt, hvilket de ville være, hvis de var hvid støj, fås at *likelihood funktionen*<sup>22</sup> ser ud på følgende måde

$$p(a_1, a_2, \dots, a_n) \propto \sigma_a^{-n} \exp\left\{-\left(\sum_{t=1}^n \frac{a_t^2}{2\sigma_a^2}\right)\right\}$$

og *log likelihood funktionen*

$$l(\phi, \theta, \sigma_a) = -n \ln \sigma_a - \frac{S(\phi, \theta)}{2\sigma_a^2} + \text{konstant}$$

hvor

$$S(\phi, \theta) = \sum_{t=1}^n a_t^2(\phi, \theta | \mathbf{w}_*, \mathbf{a}_*, \mathbf{w})$$

$S$  kaldes for *sum of squares* funktionen. Det ses af ligningen, at den eneste størrelse, der varierer, er  $S$ . Da den optræder med negativt fortegn, maksimeres log likelihood funktionen ved at finde det globale minimum for  $S$ .

For at undgå afhængigheden af startbetingelserne, indføres størrelsernes *ubetingede* værdier. En størrelses *ubetingede værdi* er den værdi, man ville forvente den ville have, alt andet lige. Det vil sige ukendte værdier<sup>23</sup> af  $a_t$  og  $e_t$  sættes til nul.

Inden der går videre med disse, er det nødvendigt at introducere en ny operator og en ny afledt tidsserie. Operatoren er defineret ved  $F^m z_t = z_{t+m}$ , og den kaldes

<sup>21</sup>I det følgende betegner fede typer vektorer.

<sup>22</sup>Likelihood funktionen angiver hvor trolig (eng. *likely*) en række af  $a_t$ 'er,  $a_1, a_2, \dots, a_n$  er i forhold til andre rækker, når  $\mathbf{w}$  er givet, men parametrene  $\phi, \theta$  varierer. Om teorien for likelihood funktioner, se for eksempel Andersen (1984)

<sup>23</sup>Det vil sige værdier af  $a_t$  og  $e_t$ , der ikke kan beregnes ud fra  $w_t$  (og dermed ud fra  $z_t$ ).

for *forward shift* operatoren. Den nye tidsserie,  $e_t$ , fremkommer ved i Ligning (27) at erstatte  $B$  med  $F$  og  $a_t$  med  $e_t$ . Således, at den kommer til at se ud på følgende måde

$$\phi(F)\Phi(F^s)\tilde{w}_t = \theta(F)\Theta(F^s)e_t$$

$e_t$  kaldes for *fejlen*.

Proceduren til at fremskaffe  $a_t$ 'erne er følgende:

1. Først beregnes  $e_t$ 'erne fra  $n$  og ned til 1, hvor alle ukendte  $e_t$ 'er sættes til nul.
2. Dernæst beregnes de  $w_t$ 'er, der skal anvendes som startbetingelser til differensligningen.
3. Til sidst er det muligt at beregne  $a_t$ 'erne.

Proceduren er skematiseret i Tabel 4. Når  $a_t$ 'erne er beregnet, er det muligt at beregne  $S$ .

Tabel 4. Skematisk fremstilling af beregningen af  $a_t$ . Beregningen følger pilen. Størrelser i kantede parenteser er forventede værdier.  $s$  er perioden for sæsonsvingningen,  $p$  og  $q$  er ordenen af henholdsvis AR- og MA-processerne.

$t$	$z_t$	$[a_t]$	$[w_t]$	$[e_t]$
$-2d - 2sD$		0		
$\vdots$		$\vdots$		
$-d - sD$		0		
$-d - sD + 1$	$z_{-d-sD+1}$	$[a_{-d-sD+1}]$	$[w_{-d-sD+1}]$	0
$-d - sD + 2$	$z_{-d-sD+2}$	$[a_{-d-sD+2}]$	$[w_{-d-sD+2}]$	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
0	$z_0$	$[a_0]$	$[w_0]$	0
1	$z_1$	$[a_1]$	$w_1$	$[e_1]$
2	$z_2$	$[a_2]$	$w_2$	$[e_2]$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$n$	$z_n$	$[a_n]$	$w_n$	$[e_n]$
$n + 1$				0
$n + 2$				0
$\vdots$				$\vdots$
$n + d + sD$				0

For den hér fundne model bliver  $S(\phi, \theta) = S(\theta_1, \theta_2, \Theta)$ . For at fremskaffe de størrelser, der er nødvendige for at beregne  $S$ , skal udtrykkene for  $e_t$ ,  $w_t$ ,  $a_t$  for den aktuelle model bruges. De er

$$e_t = w_t + \theta_1 e_{t+1} + \theta_2 e_{t+2} + \Theta e_{t+365} - \theta_1 \Theta e_{t+366} - \Theta \theta_2 e_{t+367}$$

$$w_t = e_t - \theta_1 e_{t+1} - \theta_2 e_{t+2} - \Theta e_{t+365} + \theta_1 \Theta e_{t+366} + \Theta \theta_2 e_{t+367}$$

og

$$a_t = w_t + \theta_1 a_{t-1} + \theta_2 a_{t-2} + \Theta a_{t-365} - \theta_1 \Theta a_{t-366} - \Theta \theta_2 a_{t-367}$$

Anvendes "steepest descent" på  $S$  fås, at de parametre, der minimerer denne, og dermed maksimerer log likelihood funktionen, bliver:

$$\theta_1 = 1.2705 \pm 0.0001$$

$$\theta_2 = -0.2858 \pm 0.0001$$

$$\Theta = 0.9122 \pm 0.0001$$

Det vil sige den resulterende model bliver

$$w_t = (1 - 1.27B + 0.29B^2)(1 - 0.91B^{365})a_t$$

hvor

$$w_t = \nabla^2 \nabla_{365} z_t$$

I næste afsnit vil det blive undersøgt, om denne model kan bruges, kan den det, vil den i tredje del af denne rapport blive anvendt til at lave forudsigelser med.

### 7.3 Diagnostisk undersøgelse af modellen

Spørgsmålet, vi nu ønsker at besvare, er, om den fundne model er "rigtig". Det er klart, at det er meget svært at sige om en model er rigtig eller forkert, men der findes statistiske metoder, der belyser nogle sider af en models "rigtighed". Endvidere er der udledt nogle krav i Afsnit 5, som under alle omstændigheder skal være opfyldt.

#### Stationaritet og invertibilitet

Da modellen er en MA-model er den per definition *stationær*, se Afsnit 6.2. *Invertibilitets* betingelsen, se samme afsnit, medfører, at alle rødderne i den karakteristiske ligning,  $\theta(B) = 0$ , skal ligge *udenfor* enhedscirklen. For den fundne model bliver den karakteristiske ligning

$$\theta(B) = (1 - 1.27B + 0.29B^2)(1 - 0.91B^{365}) = 0$$

det vil sige

$$(1 - 1.27B + 0.29B^2) = 0 \quad \vee \quad (1 - 0.91B^{365}) = 0$$

Den første ligning har løsningerne

$$B_1 = 3.42$$

$$B_2 = 1.02$$

der begge er reelle og større end 1, og dermed ligger udenfor enhedscirklen i det komplekse plan. Løsningerne til den anden ligning fås ved at isolere  $B^{365}$ , skrive ligningen på polær form og bruge De Moivres teorem, herved fås

$$r^{365}(\cos 365\theta + i \sin 365\theta) = \frac{1}{0.91}(\cos 2k\pi + i \sin 2k\pi), k = 0, \pm 1, \pm 2, \dots$$

Det vil sige løsningen er

$$r = \left(\frac{1}{0.91}\right)^{1/365} = 1.0002584$$

$$\theta = \frac{2k\pi}{365}$$

Anvendes Eulers formel, fås

$$B = 1.0002584e^{2ki\pi/365}, \quad k = 0, 1, \dots, 364$$

der også lige er uden for enhedscirklen. Resultatet er altså, at den fundne model både opfylder *stationaritet*s og *invertibilitet*s betingelserne.

#### Undersøgelse af residualerne

Indsættes ML-estimaterne  $(\hat{\phi}, \hat{\theta})$  i den multiple model

$$\phi(B)\Phi(B^s)\tilde{w}_t = \theta(B)\Theta(B^s)a_t \quad (28)$$

kan *residualet*,  $\hat{a}_t$ , skrives som

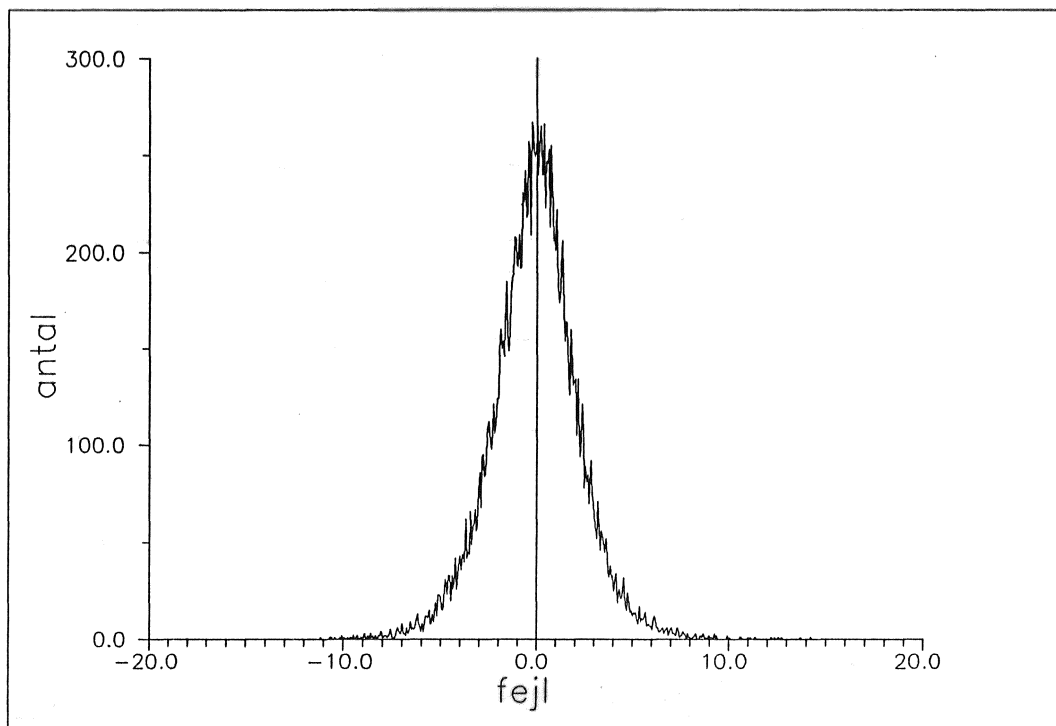
$$\hat{a}_t = \hat{\Theta}(B^s)^{-1}\hat{\theta}(B)^{-1}\hat{\phi}(B)\hat{\Phi}(B^s)\tilde{w}_t$$



Man kan vise, at

$$\hat{a}_t = a_t + O\left(\frac{1}{\sqrt{n}}\right)$$

hvor  $a_t$  er hvid støj. Det vil sige jo flere observationer, der indgår i serien, jo mere bliver  $\hat{a}_t$  som hvid støj. Derfor er det at forvente, at hvis  $\hat{a}_t$  afviger fra hvid støj, så er der en utilstrækkelighed ved modellen. Studerer man Figur 32, ser man, at residualerne faktisk er normalfordelt omkring 0. Beregnes middelværdien af residualerne fås den til at være:  $-5.93 \cdot 10^{-2}$ , spredningen er 2.28.



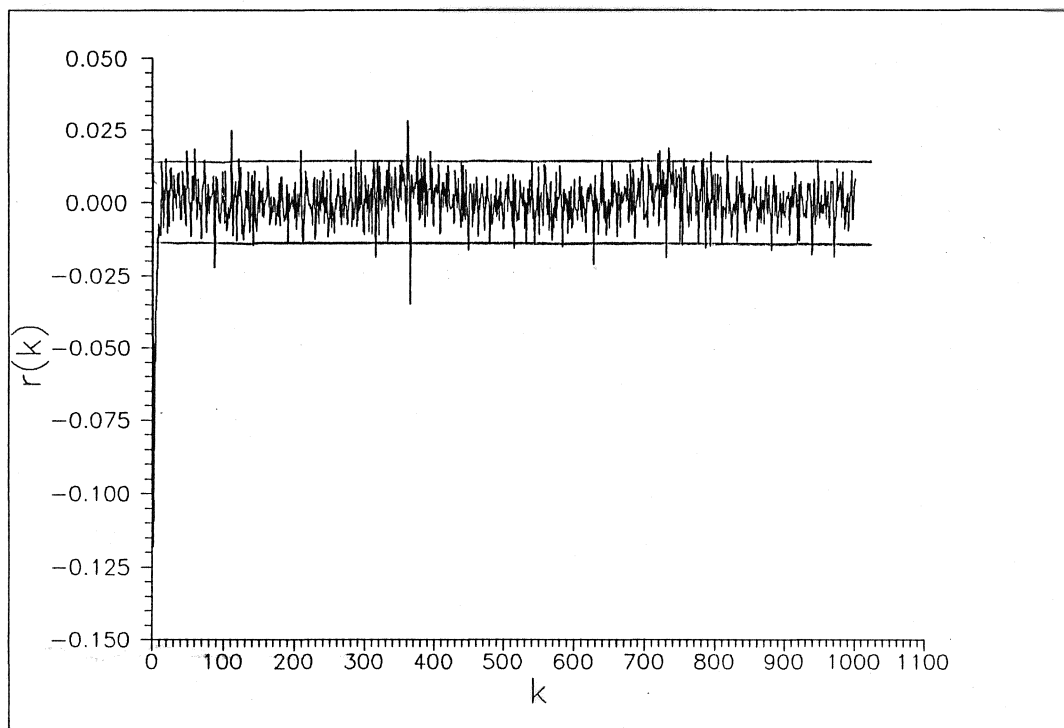
Figur 32. Fordelingen af residualen,  $a_t$ . Bemærk, at ifølge side 66 er  $a_t$  også et udtryk for fejlen. Fordelingen viser antallet af residualer, der ligger indenfor et interval, der er 0.06 bredt. Punktet, der angiver antallet som funktion af fejlen er placeret midt i intervallet.

**Undersøgelse af autokorrelationerne** Havde vi de virkelige parametre i modellen,  $(\phi, \theta)$ , har Anderson, 1942 vist, at autokorrelationerne for residualen,  $r_k(a)$ , er normalfordelt med middelværdi 0 og spredning  $\sigma_r = n^{-1/2}$ . Da det imidlertid er sådan, at vi kun kender ML-estimerne  $(\hat{\phi}, \hat{\theta})$ , gælder der i følge Box og Pierce (Box and Pierce, 1970), at  $n^{-1/2}$  blot er en øvre grænse for  $\sigma_r$ . På Figur 33 er autokorrelationerne tegnet ind sammen med denne øvre grænse. For  $n = 20,000$  er  $\sigma_r = 7.1 \cdot 10^{-3}$ . Som det ses af denne er der nogle autokorrelationer, der ligger over denne grænse. Dette gør imidlertid intet, for beregner man spredningen af de fundne residualers autokorrelationer, er den  $7.4 \cdot 10^{-3}$ , og gennemsnittet er  $6.7 \cdot 10^{-5}$ , hvilket stemmer nøje overens med de ifølge teorien forventede værdier.

Istedet for at undersøge autokorrelationerne enkeltvis, har Box og Pierce (Box and Pierce, 1970) vist, at størrelsen

$$Q = n \sum_{k=1}^K r_k^2(\hat{a})$$

er fordelt som  $\chi^2(K-p-P-q-Q)$ .  $K$  er det tal, hvorom der gælder at koefficienterne,  $\psi_j$ , i udtrykket  $\hat{w}_t = \psi(B)$  er negligeable for  $j > K$ ,  $n = N - d - sD$  er antallet af



Figur 33. Autokorrelationerne af residualet for  $k = 1$  til 10,000. Linien over og under  $x$ -aksen er den øvre grænse.

differencer dannet ud fra tidsserien. For den fundne model er  $K = 367$ , og  $Q$  skal være fordelt som  $\chi^2(367 - 0 - 0 - 2 - 1 = 364)$ .  $Q = 1,056.2$ , svarer til en percentil på 100.0 for den førnævnte fordeling<sup>24</sup>. Dette betyder, under forudsætning af at hypotesen er korrekt, at der i 0% af tilfældene kan forventes afvigelser større end de her fundne, eller med andre ord, at modellen må forkastes. At vi alligevel ikke forkaster modellen skyldes, at  $n$  er (for) stor, er  $n$  for eksempel 6,000, finder man, at størrelserne af autokorrelationerne er de samme, men værdien af  $Q$  gør, at man ville forvente afvigelser større end den her fundne i 100% af tilfældene. Dermed ville det have været en (meget) god model (se iøvrigt diskussionen i næste afsnit).

**Det kumulative periodogram** For en endelig serie af hvid støj gælder der, at spektrummet,  $p(f)$ , har en konstant værdi,  $2\sigma_a^2$ , for frekvenser mellem 0 og 0.5 cykler, det vil sige for  $f = 0/N$  til  $f = N/2/N$ . Dette medfører, at  $P(f)$ , defineret ved

$$P(f) = \int_0^f p(g) dg$$

tegnet mod  $f$  danner en ret linie fra  $(0, 0)$  til  $(0.5, \sigma_a^2)$ . Divideres  $P(f)$  med  $\sigma_a^2$ , går den rette linie fra  $(0, 0)$  til  $(0.5, 1)$ . Da  $I(f_i)$  estimerer  $P(f)$  er  $1/n \sum_{i=1}^j I(f_i)$  et estimat af  $P(f)$  og dermed er

$$C(f_i) = \frac{\sum_{j=1}^i I(f_j)}{ns^2}$$

et estimat af  $p(f_i)/\sigma_a^2$  og  $s^2$  er et estimat af  $\sigma_a^2$ . Kolmogorov og Smirnov (se Hald, 1952) har givet nogle grænser for, hvor meget linien dannet af  $C(f_i)$ ,  $f_i$  må afvige

<sup>24</sup> Dette tal er fundet ved numerisk at integrere

$$\Phi(x) = \frac{1}{2^{364/2} \Gamma(364/2)} \int_0^x t^{(364-2)/2} e^{-t/2} dt$$

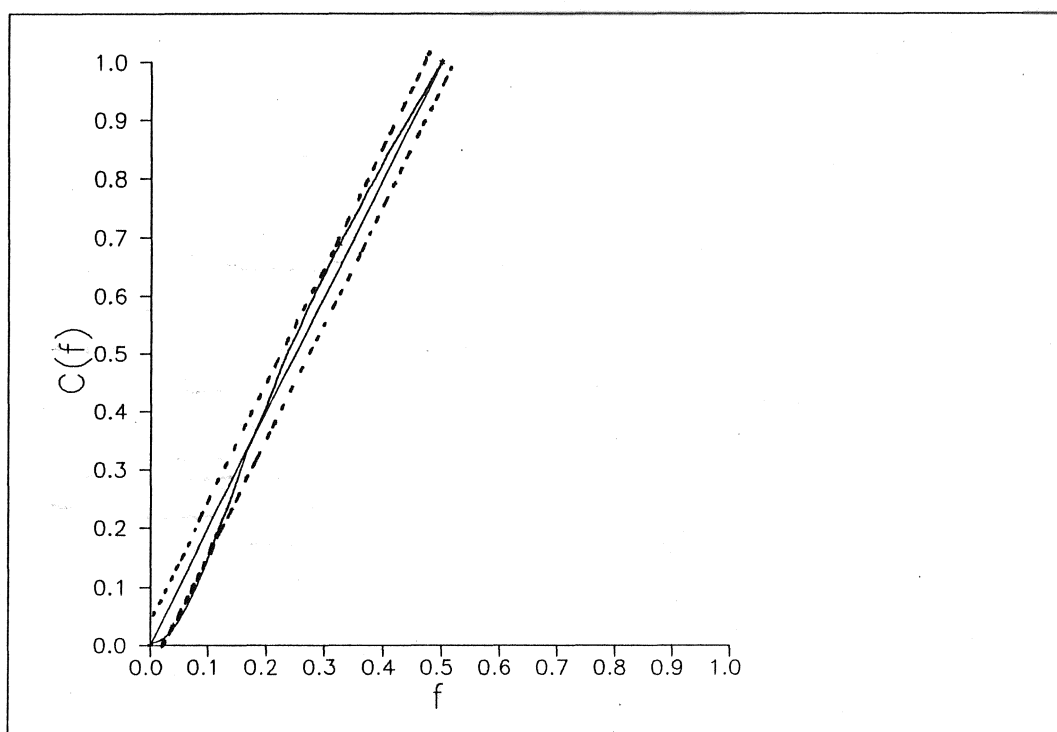
da det ikke har været muligt at finde tabeller, der dækker frihedsgrader af denne størrelse.

fra den teoretiske: Der gælder, at en  $\epsilon$ -del af punkterne må ligge i en afstand større end  $\pm K_\epsilon/\sqrt{q}$  fra linien, der går fra (0,0) til (0.5,1). Værdien af  $K_\epsilon$  er tabuleret i Tabel 5 og  $q = (n - 2)/2$  for  $n$  lige og  $q = (n - 1)/2$  for  $n$  ulige.

Tabel 5. Kolmogorov-Smirnov parametrene  $K_\epsilon$ .  $\epsilon$  angiver hvor stor en del af de kumulerede intensiteter, der må ligge udenfor linien.

$\epsilon$	0.01	0.05	0.10	0.25
$K_\epsilon$	1.63	1.36	1.22	1.02

På Figur 34 er det kumulerede periodogram vist sammen med linien  $\epsilon = 0.01$ . Som det ses, falder periodogrammet næsten indenfor de to linier. Studerer man figuren nøje, ser man, at der ligger cirka 5% af punkterne udenfor linien. Dette kan have flere forklaringer, en af dem er den store mængde data, der er anvendt. Dette medfører, at  $q$  bliver stor, og dermed bliver afstanden fra linien lille. Prøver man at tage færre data (for eksempel 10,000), er det kumulative periodogram det samme, blot bliver afstanden,  $K_\epsilon/q$ , større, således at alle punkterne falder indenfor. Det viser sig også, at det er uden betydning for udsvinget af linien, som punkterne danner, hvilken model, der anvendes. Prøver man for eksempel med en  $ARIMA(0, 1, 1) \times ARIMA(0, 1, 1)_{365}$ , er billedet præcist det samme. Så fejlen ligger ikke i, at der ikke er taget tilstrækkelig hensyn til den årlige variation. En konklusion, man kunne drage af dette, er, at det ikke er nødvendigt for ARIMA-processen at have kendskab til så mange data, for det eneste processen 'får ud af' data'ene er, at der er en sæsonsvingning overlejret med støj. Dette (måske) til forskel fra de neurale netværk, der lærer at simulere støjen bedre, jo flere observationer de bliver trænet i.



Figur 34. Det kumulerede periodogram vist sammen med linien for  $\epsilon = 0.01$ . Den rette linie i midten er den teoretiske kurve.

**Konklusion** Da ingen af de anvendte tests har kunnet "slå modellen af pinden", anvendes den i næste del til at lave forudsigelser med. Der har dog været problemer på grund af den store datamængde. Bemærk dog, at der sagtens kan være alt muligt andet galt med modellen, det er bare hér sandsynliggjort, at den er brugbar.

# **Part III**

## **Forudsigelse**





## 8 Indledning

I denne tredje og sidst del af rapporten vil der blive afholdt en "konkurrence" mellem de neurale netværk, der blev fundet i 1. del, og ARIMA-modellen udviklet i 2. del. "Konkurrencen" går ud på at se hvilken af de to metoder, der bedst forudsiger temperaturserien henholdsvis 1 og 5 døgn frem. Altså en dag-til-dag forudsigelse og en 5-døgns prognose.

"Reglerne" for konkurrencen er, at hver metode får temperaturserien givet op til tid  $t$ , derefter skal værdien af temperaturserien til henholdsvis  $t + 1$  og  $t + 5$  forudsiges. Dette gøres for 10,000 observationer og fejlene sammenlignes<sup>25</sup>.

### 8.1 Forudsigelse ved hjælp af ARIMA-processen

I dette afsnit vil teorien for forudsigelse ved hjælp af en ARIMA-proces blive gennemgået og den i 2. del fundne model vil blive brugt til at forudsige med.

#### Mindste kvadraters fejl-forudsigelser

En multipel ARIMA-model for  $z$  til tiden  $t + l$

$$\phi(B)\Phi(B^s)\nabla^d\nabla_s^D z_{t+l} = \theta(B)\Theta(B^s)a_{t+l}$$

kan skrives på 3 forskellige måder

1. Direkte som differensligning

$$w_{t+l} = \theta(B)\Theta(B^s)a_{t+l} + f(B)w_{t+l} + F(B^s)(1 - f(B))w_{t+l}$$

hvor  $w_{t+l} = \nabla^d\nabla_s^D z_{t+l}$ ,  $f(B) = \phi_1 B + \phi_2 B^2 + \dots + \phi_p B^p$  og  $F(B^s) = \Phi_1 B^s + \Phi_2 B^{2s} + \dots + \Phi_P B^{Ps}$ .

2. Som en uendelig sum af chok  $a_j$

$$z_{t+l} = \sum_{j=-\infty}^{t+l} \psi_{t+l-j} a_j = \sum_{j=0}^{\infty} \psi_j a_{t+l-j}$$

hvor  $\psi_0 = 1$

3. Som en uendelig sum af tidligere observationer plus et chok

$$z_{t+l} = \sum_{j=1}^{\infty} \pi_j z_{t+l-j} + a_{t+l}$$

ønskes det at lave en forudsigelse,  $\hat{z}_t(l)$ , af  $z_t$  til tiden  $t+l$ , der skal være en lineær sum af nuværende og tidligere observationer,  $z_t, z_{t-1}, \dots$ , så vil forudsigelsen også være en lineær sum af nuværende og tidligere chok  $a_t, a_{t-1}, \dots$

Antag da, at den bedste forudsigelse er

$$\hat{z}_t(l) = \psi_l^* a_t + \psi_{l+1}^* a_{t-1} + \psi_{l+2}^* a_{t-2} + \dots$$

hvor vægtene  $\psi_l^*, \psi_{l+1}^*, \dots$  skal bestemmes. Gennemsnittet af kvadratet på fejlen er da givet ved

$$E[z_{t+l} - \hat{z}_t(l)]^2 = E\left[\sum_{i=0}^{\infty} \psi_i a_{t+l-i} - \sum_{j=0}^{\infty} \psi_{l+j}^* a_{t-j}\right]^2$$

<sup>25</sup> Det kan være svært at udnævne en "vinder" i alle situationer, da der anvendes flere forskellige mål for fejlen.

hvor den multiple model er skrevet som en uendelig sum af  $a_t$ . Omindekseres den første sum til  $k = t+l-i$ , kan fejlen skrives

$$E\left[\sum_{k=-\infty}^{t+l} \psi_{t+l-k} a_k - \sum_{j=0}^{\infty} \psi_{l+j}^* a_{t-j}\right]^2$$

Splittes den første sum op i to, er dette lig

$$E\left[\sum_{k=-\infty}^t \psi_{t+l-k} a_k + \sum_{k=t+1}^{t+l} \psi_{t+l-k} a_k - \sum_{j=0}^{\infty} \psi_{l+j}^* a_{t-j}\right]^2$$

reindekseres den første sum til  $j = t-k$ , fremkommer

$$E\left[\sum_{j=0}^{\infty} \psi_{l+j} a_{t-j} + \sum_{k=t+1}^{t+l} \psi_{t+l-k} a_k - \sum_{j=0}^{\infty} \psi_{l+j}^* a_{t-j}\right]^2$$

samles første og sidste sum til én, fås

$$E\left[\sum_{j=0}^{\infty} (\psi_{l+j} - \psi_{l+j}^*) a_{t-j} + \sum_{k=t+1}^{t+l} \psi_{t+l-k} a_k\right]^2$$

Kvadreres udtrykket og midles de enkelte led, fremkommer

$$E\left[\sum_{j=0}^{\infty} (\psi_{l+j} - \psi_{l+j}^*) a_{t-j}\right]^2 + E\left[\sum_{k=t+1}^{t+l} \psi_{t+l-k} a_k\right]^2 + 2E\left[\sum_{j=0}^{\infty} (\psi_{l+j} - \psi_{l+j}^*) a_{t-j} \sum_{k=t+1}^{t+l} \psi_{t+l-k} a_k\right]$$

Da

$$E[a_t a_{t+k}]^2 = \begin{cases} \sigma_a^2 & k = 0 \\ 0 & k \neq 0 \end{cases}$$

bliver middelværdien af kvadratet på fejlen endelig<sup>26</sup>

$$E[z_{t+l} - \hat{z}_t]^2 = (1 + \psi_1^2 + \dots + \psi_{l-1}^2) \sigma_a^2 + \sum_{j=0}^{\infty} (\psi_{l+j} - \psi_{l+j}^*)^2 \sigma_a^2$$

Det ses nu, at skal fejlen minimeres, så må  $\psi_{l+j} = \psi_{l+j}^*$ , for alle  $j$ . Dette resultat er vigtigt, da det siger, at den bedste forudsigelse fremkommer ved at anvende den for den historiske tidsserie fundne model.

Anvendes det fundne resultat, kan værdien af tidsserien til tid  $t+l$  skrives

$$\begin{aligned} z_{t+l} &= (a_{t+l} + \psi a_{t+l-1} + \dots + \psi_{l-1} a_{t+1}) + (\psi_l a_t + \psi_{l+1} a_{t-1} + \dots) \\ &= e_t(l) + \hat{z}_t(l) \end{aligned}$$

hvor  $e_t(l)$  er fejlen ved forudsigelsen  $\hat{z}_t(l)$  til tiden  $t+l$ .

Variansen af fejlen er givet af

$$V(l) = \text{var}[e_t(l)] = (1 + \psi_1^2 + \dots + \psi_{l-1}^2) \sigma_a^2$$

Beregnes  $e_t(1)$  fås

$$e_t(1) = z_{t+1} - \hat{z}_t(1) = a_{t+1}$$

Af dette ses, at  $a_{t+1}$ , foruden at kunne betragtes som hvid støj, også er fejlen ved forudsigelsen ét tidsskridt frem.

### Anvendelse af differensmetoden til forudsigelse

Benyttes det faktum, der blev udledt i det sidste afsnit, at den bedste forudsigelse fås ved at anvende parametrene fra modellen for den historiske tidsserie, kan man skrive forudsigelsen på de tre måder, der også blev nævnt i sidste afsnit. I dette

<sup>26</sup>Det sidste led i den foregående ligning er 0, da  $a_t$ 'erne i de to summer ikke "overlapper".

afsnit udledes forudsigelsen v.hj.a. differensmetoden. Forudsigelsen af  $w$  til tiden  $t+l$ ,  $\hat{w}_t(l)$ , er givet ved

$$\hat{w}_t(l) = \theta(B)\Theta(B^s)[a_{t+l}] + f(B)[w_{t+l}] + F(B^s)(1-f(B))[w_{t+l}]$$

Det, at en størrelse er i kantede parenteser, betyder, at værdien er den forventede værdi under betingelse af den givne tidsserie,  $z_t, z_{t-1}, \dots$ . De forventede værdier er givet ved

$$\begin{aligned} [z_{t+j}] &= \begin{cases} z_{t+j} & j \leq 0 \\ \hat{z}_t(j) & j > 0 \end{cases} \\ [a_{t+j}] &= \begin{cases} a_{t+j} & j \leq 0 \\ 0 & j > 0 \end{cases} \end{aligned} \quad (29)$$

Anvendes dette på den i 2. del fundne multiple ARIMA-model

$$z_t = 2z_{t-1} - z_{t-2} + z_{t-365} - 2z_{t-366} + z_{t-367} + (1 - \theta_1 B - \theta_2 B^2)(1 - \Theta B^{365})a_t$$

fås at forudsigelsen til tiden  $t+l$  er givet ved

$$\begin{aligned} \hat{z}_t(l) &= 2[z_{t+l-1}] - [z_{t+l-2}] + [z_{t+l-365}] - 2[z_{t+l-366}] + [z_{t+l-367}] \\ &\quad + (1 - \theta_1 B - \theta_2 B^2)(1 - \Theta B^{365})[a_{t+l}] \end{aligned} \quad (30)$$

ønskes det at lave forudsigelse ét tidsskridt frem, fremkommer følgende udtryk, ved at anvende Ligning (29)

$$\begin{aligned} \hat{z}_t(1) &= 2z_t - z_{t-1} + z_{t-364} - 2z_{t-365} + z_{t-366} - \theta_1 a_t - \theta_2 a_{t-1} - \Theta a_{t-364} + \Theta \theta_1 a_{t-365} \\ &\quad + \Theta \theta_2 a_{t-366} \end{aligned}$$

hvor  $[a_{t+1}]$  er sat lig 0. Forudsigelsen 5 døgn frem findes ved at sætte  $l = 5$  i Ligning (30), herved fremkommer

$$\hat{z}_t(5) = 2\hat{z}_t(4) - \hat{z}_t(3) + z_{t-360} - 2z_{t-361} + z_{t-362} - \Theta a_{t-360} + \Theta \theta_1 a_{t-361} + \Theta \theta_2 a_{t-362}$$

$\hat{z}_t(4)$  og  $\hat{z}_t(3)$  beregnes ved at sætte  $l$  lig henholdsvis 4 og 3 i Ligning (30). Gøres dette, finder man, at det også er nødvendigt at beregne  $\hat{z}_t(2)$  og  $\hat{z}_t(1)$ , dette gøres på samme måde ved at sætte  $l$  lig 2 i (30), 1 døgnns prognosen er beregnet ovenfor.

**Resultater af anvendelse af modellen til forudsigelse** Anvendes modellen til at forudsige henholdsvis 1 og 5 døgn frem på 10,000 observationer, der ikke har været anvendt i udviklingen af modellen, fås resultaterne vist i henholdsvis Tabel 6 og 7.

*Tabel 6. Middelværdi (i °C), spredning (i °C), relativ værdi, korrelations koeficient, index, minimum (i °C) og maximum (i °C) af fejlen for alle modellerne anvendt til at forudsige temperaturserien 1 døgn frem. Der er anvendt 10,000 observationer til beregningen. De beregnede størrelser er defineret i Afsnit 4.2.*

1 døgn	ARIMA	net1b	net2b	net2b*	Persistens
Middelværdi	$-3.18 \cdot 10^{-3}$	$-2.01 \cdot 10^{-2}$	$-2.31 \cdot 10^{-2}$	$-6.78 \cdot 10^{-3}$	$5.80 \cdot 10^{-4}$
Spredning	2.35	2.31	2.33	2.33	2.27
Relativ værdi	$-5.61 \cdot 10^{-2}$	$-4.58 \cdot 10^{-2}$	$-2.96 \cdot 10^{-2}$	$-2.29 \cdot 10^{-2}$	$5.08 \cdot 10^{-2}$
Korrel. koef.	0.97	0.98	0.97	0.97	0.97
Index	0.34	0.34	0.34	0.34	0.35
Minimum	-14.0	-16.6	-16.8	-16.9	-11.3
Maximum	15.3	14.4	14.1	13.9	16.5



Tabel 7. Middelværdi ( $i^{\circ}C$ ), spredning ( $i^{\circ}C$ ), relativ værdi, korrelations koefficient, index, minimum ( $i^{\circ}C$ ) og maksimum ( $i^{\circ}C$ ) af fejlen for alle modellerne anvendt til at forudsige temperaturserien 5 døgn frem. Der er anvendt 10,000 observationer til beregningen. De beregnede størrelser er defineret i Afsnit 4.2.

5 døgn	ARIMA	net1b	net2b	net2b*	Persistens
Middelværdi	$-1.22 \cdot 10^{-2}$	$-4.46 \cdot 10^{-2}$	$-5.95 \cdot 10^{-2}$	$-3.02 \cdot 10^{-3}$	$2.95 \cdot 10^{-3}$
Spredning	3.60	2.40	2.39	2.40	3.49
Relativ fejl	$-1.01 \cdot 10^{-1}$	$-6.94 \cdot 10^{-2}$	$-1.84 \cdot 10^{-2}$	$-3.65 \cdot 10^{-2}$	$1.73 \cdot 10^{-1}$
Korrel. koef.	0.94	0.97	0.97	0.97	0.94
Index	0.53	0.35	0.35	0.35	0.53
Minimum	-17.4	-13.8	-14.6	-14.5	-16.5
Maximum	15.5	15.7	14.9	15.5	16.7

Bruger man for eksempel en ARIMA(0,0,1) $\times$ (0,1,1)<sub>365</sub>-model<sup>27</sup> for at sammenligne to forskellige ARIMA-modeller, findes at der er overensstemmelse mellem de to modeller, men at den i afsnittet om modeludvikling fundne er bedst.

Det er interessant at se, om man kan bruge ARIMA-processen til at lave "klima-modeller" med, det vil sige til at forudsige temperaturen langt mere end en måned frem. Anvendes outputtet fra ARIMA-modellen som nyt input til den 1,000 gange, fås resultatet vist i Figur 35. Som det ses af denne, kan forudsigelsen kun betragtes som brugbar cirka 30 døgn frem<sup>28</sup>, hvilket da også må siges at være godt, men ikke godt nok til en klimamodel. Bemærk, at selvom prognosen fjerner sig meget fra det, den skal prognosticere, så har den den årlige svingning med. Bemærk også, at prognosen fjerner sig *lineært* fra temperaturserien, der er altså tale om en akkumulering af fejlen. Box, se Box and Jenkins (1970) appendix A5.1, nævner da også, at fejlene,  $a_t$ , for prognosen er korrelerede - skyder den første prognose over, så skyder resten også over.

## 8.2 Forudsigelse ved hjælp af neurale netværk

Når et neuralt netværk skal anvendes til at forudsige med, er metoden den, som nævnt i 1.del, at man oplærer netværket i map'en og derefter præsenterer det oplærte netværk for den del af tidsserien, det skal forudsige. De net, der skal bruges til at forudsige med, er net 1b og net 2b fra 1. del. b versionen, den med direkte forbindelse fra inputlaget til outputlaget, er valgt, fordi det ønskes, at nettet også får lineære egenskaber.

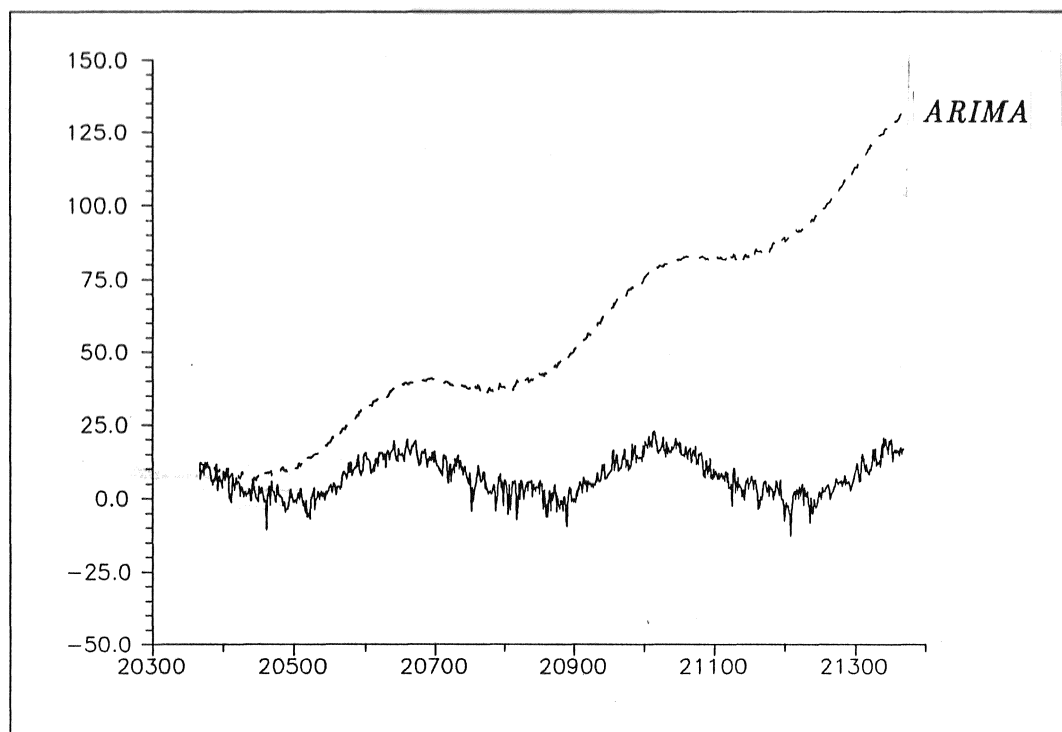
Input'et til netværk 1b er  $z_t$  og det ønskede output er  $z_{t+1}$  for 1-døgns prognosen og  $z_{t+5}$  for 5-døgns prognosen. Netværk 2b anvendes i to versioner: én hvor input'et er  $z_{t-3}, z_{t-2}, z_{t-1}, z_t$  og det ønskede output er henholdsvis  $z_{t+1}$  og  $z_{t+5}$ , dette net betegnes 2b. I den anden version er output'et det samme ( $z_{t+1}$  eller  $z_{t+5}$ ), men input'et er  $z_{t-365}, z_{t-364}, z_{t-1}, z_t$ , for at se om inkluderingen af værdier en periode tilbage forbedrer forudsigelsen, dette net betegnes 2b\*.

En map for 1-døgns prognosen, ( $z_t, z_{t+1}$ ), er vist i Figur 36 og for 5-døgns prognosen, ( $z_t, z_{t+5}$ ), i Figur 37. Da map'erne ikke er streger, men udtværede punktmængder, ses det, at for at "imbedde" tidsserien er det nødvendigt at gøre imbedding-dimensionen større end 2.

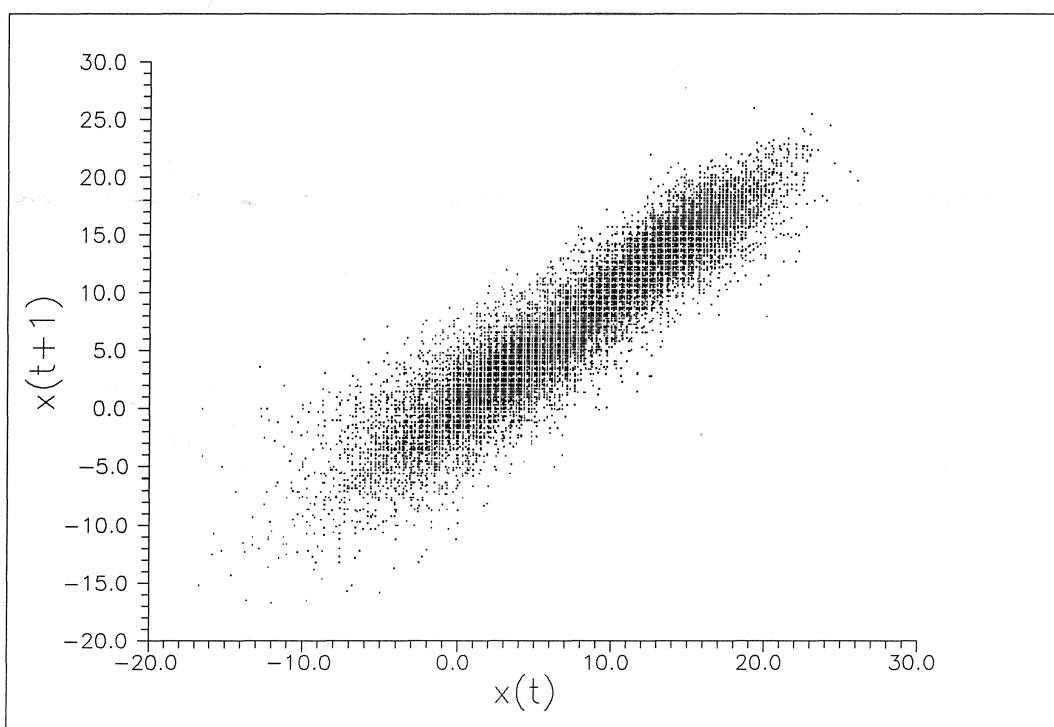
Netværkene optrænes i henholdsvis dag-til-dag forudsigelsen og 5-døgns prognosen, ved at de 20,000 første værdier af temperaturserien bliver præsenteret 2,500

<sup>27</sup> Parametrene for denne model er:  $\theta = 0.55$  og  $\Theta = 0.92$ .

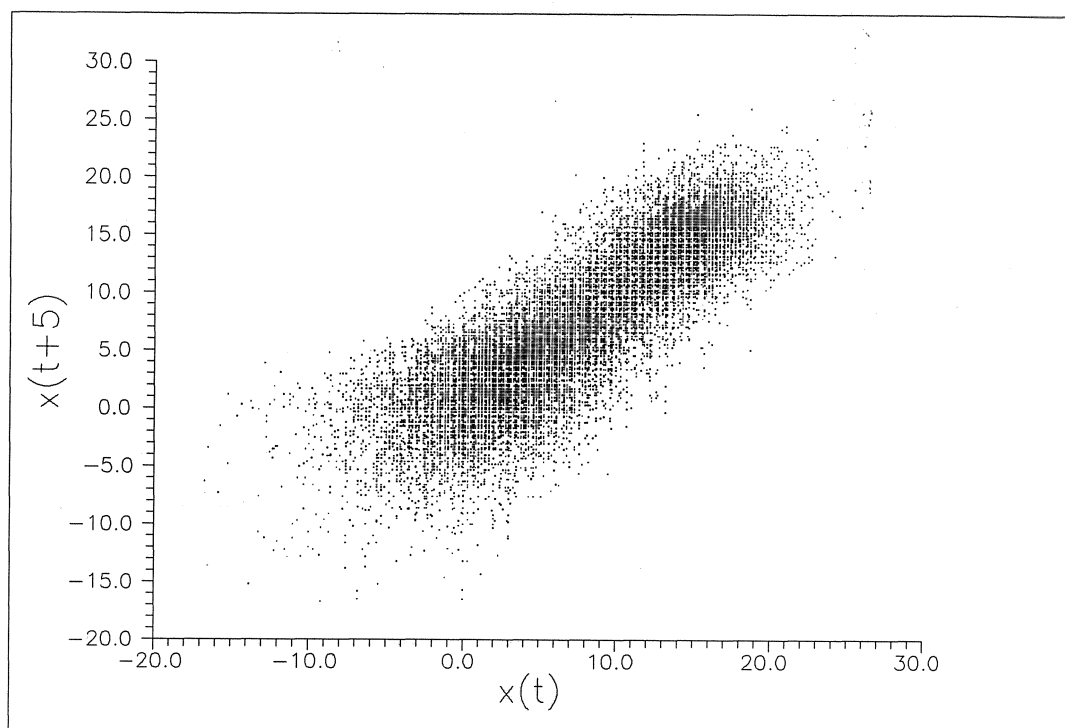
<sup>28</sup> Den er dog kun brugbar i den forstand, at den følger den årlige svingning, helt fra den første dag følger den ikke de daglige ændringer.



Figur 35. Forudsigelse af temperaturen 1,000 døgn frem ved hjælp af ARIMA-modellen. Der er kun anvendt kendte værdier af temperaturserien, det vil sige værdier op til  $t=20366$ .



Figur 36. Plot af den målte temperaturseries map givet ved:  $(z_t, z_{t+1})$ .



Figur 37. Plot af den målte temperaturseries map givet ved:  $(z_t, z_{t+5})$ .

gange for nettene. Der er anvendt 20,000 værdier for at se om der er et eller andet i støjen som netværkene kan lære. Dette kunne for eksempel være en form for deterministisk kaos. Anvendes de oplærte netværk til at forudsige ét og fem døgn frem for de næste 10,000 værdier, fås resultaterne vist i henholdsvis Tabel 6 og Tabel 7. Af disse fremgår det, at det ikke kan betale sig at tage værdier med en periode tilbage, da man ikke vinder noget i nøjagtighed. Sammenlignes resultaterne fra de neurale netværk med de andre metoder, ser det ikke ud som om at netværkene har "set" noget som de andre ikke har. Dette betyder at den egenskab, der dominerer temperaturserien er den periodiske svingning.

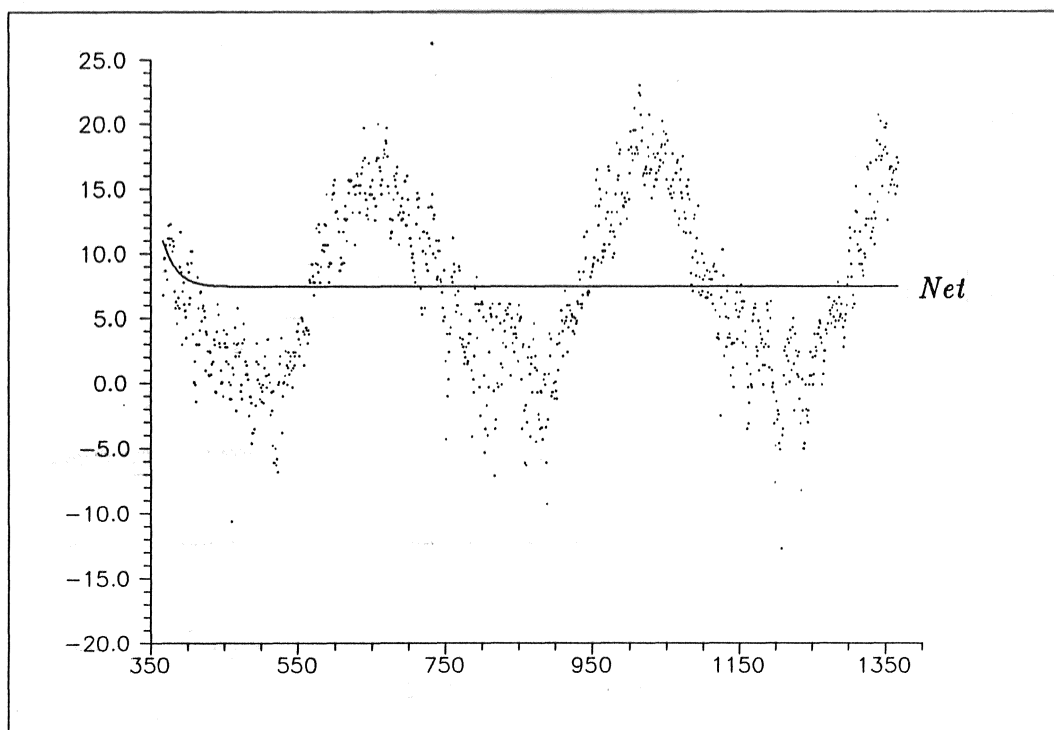
Studerer man vægtene for alle de oplærte netværk, er der én ting der tydeligt træder frem, det er, at værdien af vægten af input'et fra  $z_t$  til outputneuronen er større end de andre vægte. Det vil sige, at de neurale netværk alle har fundet ud af, at persistens spiller en stor rolle i korttids forudsigelserne.

Anvendes de neurale netværk, der er trænet til 1-døgns prognosen, til at lave langtidsprognoser med, fås resultatet vist i Figur 38. Som det ses af denne laver nettene en anden slags langtidsprognose end ARIMA-modellen, idet nettene går i en stabil tilstand (et fix-punkt ?) og bliver der. På figuren er resultatet for net2b\* vist, de andre net er helt magen til. Bemærk at årstidsvariationen helt mangler – også for net 2b\* – men at fejlen til gengæld ikke vokser. Konkluderede kan man sige, at heller ikke de neurale netværk, optrænet til at lave 1-døgns prognoser, kan anvendes til at lave klimamodeller med, dermed være ikke sagt at man ikke kan bruge neurale netværk til at lave klimamodeller overhovedet, de skal måske bare trænes til det.

I denne forbindelse er det værd at nævne at atmosfærens forudsigelighed er begrænset. Dette fandt Lorenz (Lorenz, 1963) ud af allerede i 1960'erne, idet han sagde, at på grund af at de ligninger man bruger til forudsigelser kun er approksimationer<sup>29</sup>, og begyndelsesbetingelserne altid vil være forbundet med en

<sup>29</sup> Tilmed ikke-lineære.

vis usikkerhed, vil der altid være en *indbygget* begrænsning for længden af brugbare prognoser. Størrelsen af denne øvre grænse er der nogen usikkerhed forbundet med, Smagorinsky (Smagorinsky, 1969) har fundet, at fejlen vokser eksponentielt, med en fordoblingstid på 2.5 dage og Lorenz selv fandt, se Lorenz, (1969), at der er en delvis forudsigelighed op til mindst 12 dage.



Figur 38. Forudsigelse af temperaturen 1,000 døgn frem ved hjælp af det neurale netværk 2b\*. Når først forudsigelsen er startet er der ikke anvendt værdier fra den målte temperaturserie. Den fuldt optrukne linie er forudsigelsen og punkterne er målingerne.

### 8.3 Persistens

Da en af de ting, man kan lære af de neurale netværk, var, at temperaturen til tid  $t$  er en god approksimation til temperaturen til tid  $t + 1$  og også til  $t + 5$ , er det interessant at prøve med disse to *persistensmodeller*. Persistensmodellen<sup>30</sup> for dag-til-dag forudsigelsen er

$$z_{t+1} = z_t$$

og for 5-døgns prognosen

$$z_{t+5} = z_t$$

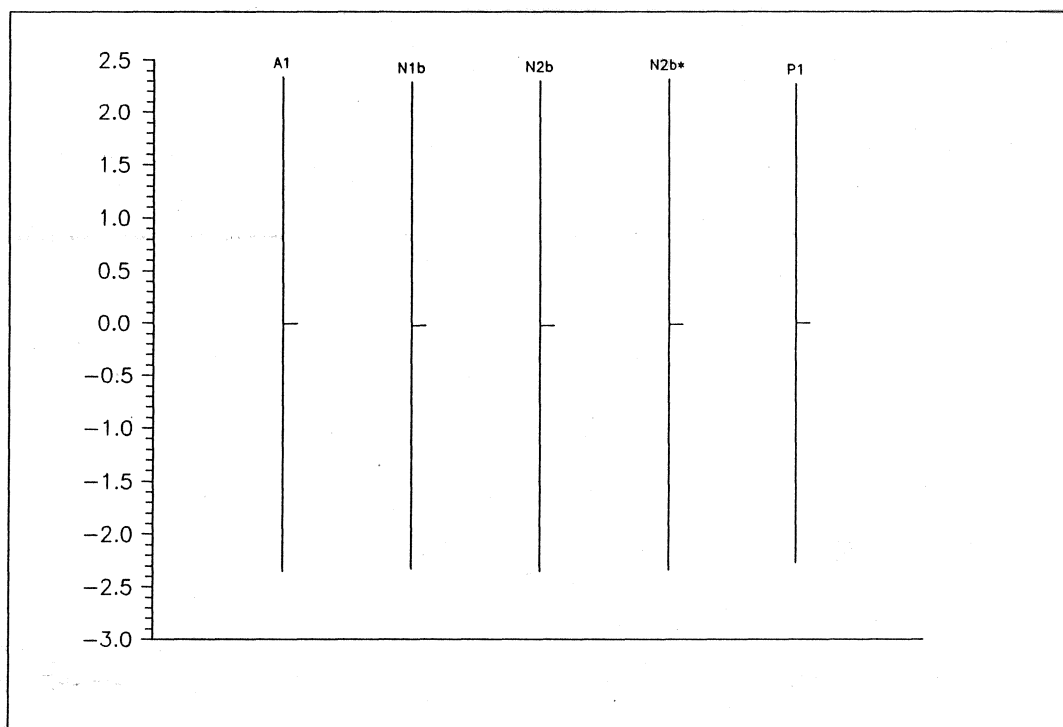
Anvendes disse to modeller fås resultaterne vist i Tabel 6 og 7. Som det ses af disse, er det en *meget* god forudsigelse at sige, at "vejret i morgen bliver ligesom i dag".

<sup>30</sup>Persistensmodeller kan også udtrykkes i ARIMA-sproget. 1-døgns prognosen er en ARIMA(0,0,1) model og 5-døgns prognosen er en ARIMA(0,0,1)<sub>5</sub>-model. I neuralt netværks sprog er persistensmodellen givet ved at alle vægte er store og negative (hæmmende), undtagen vægten mellem input-laget og output-laget, der er 1. Alle tærskler er nul.



## 9 Konklusion

Af tabellerne 6 og 7 fremgår det, at der ikke er den store forskel på de forskellige metoder (inklusive persistensmodellen), da næsten alle fejlmålene er lige store. Dette er tydeliggjort i Figur 39 hvor middelværdien plus/minus spredningen af fejlen for 1-døgns prognosen er tegnet for de 5 forskellige forudsigelser. Det ses også af tabellerne og Figur 40, at når der er tale om 5-døgns prognoser, så er de neurale netværk en del bedre end de to andre metoder, men stadig er metodernes nøjagtighed sammenlignelige. Nogle ville måske hævde, at det er "snyd" at sammenligne de neurale netværk, der netop er optrænet i 5-døgns prognoserne, imod ARIMA-modellen. Dertil er at sige, at ARIMA-modellerne er konstrueret sådan, at den samme model skal bruges til *alle* forudsigelser, selvfølgelig bliver fejlen større, jo længere frem forudsigelsen rækker.

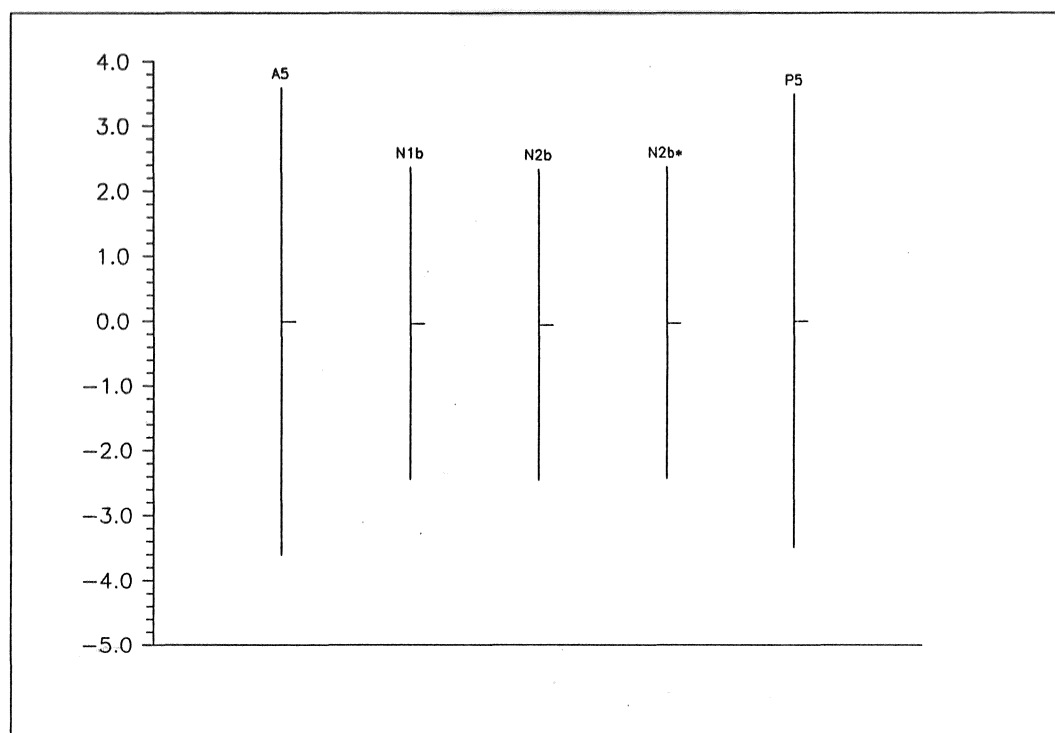


Figur 39. Sammenligning af middelværdi og spredning for de 5 forskellige forudsigelser 1 døgn frem. Den vandrette streg midt i den lodrette er middelværdien og den lodrette er 2 gange spredningen lang. A1 er ARIMA-modellen anvendt til at forudsige temperaturserien 1 døgn frem. N1b er netværk 1b, N2b er netværk 2b og N2b\* er netværk 2b\*. P1 er persistensmodellen.

Det er interessant at prøve at betragte de to metoder som "black boxes". Gøres dette, viser det sig, at de har nogle ligheder: begge metoder har værdier fra tidsserien som input og forudsigelser af fremtidige værdier som output. I begge metoderne indgår også parametre, i ARIMA-modellen  $\theta$ 'erne og i de neurale netværk vægtene og tærsklerne, der skal varieres sådan at fejlen metoden begår bliver så lille som mulig.

åbner man det neurale netværks "black box", ses det, at outputtet blot er en funktion af inputtet. Funktionen er for net 1b givet ved

$$(z_{t+1})O_6 = \sum_{j=1}^5 V_{6j}(0.5(1 + \tanh(V_{j0}z_t - T_j))) + V_{6,0}z_t - T_6$$



Figur 40. Sammenligning af middelværdi og spredning for de 5 forskellige forudsigelser 5 døgn frem. Den vandrette streg midt i den lodrette er middelværdien og den lodrette er 2 gange spredningen lang. A5 er ARIMA-modellen anvendt til at forudsige temperaturserien 5 døgn frem. N1b er netværk 1b, N2b er netværk 2b og N2b\* er netværk 2b\*. P5 er persistensmodellen.

så når man anvender neurale netværk til at forudsige tidsserier med, er der tale om en approksimation af tidsserien med en funktion.

Konkluderende kan man sige, at neurale netværk til at forudsige tidsserier (selv kaotiske !) er kommet for at blive. For det første er anvendelsen simpel, da man ikke behøver at foretage lange analyser af data for at finde det rette netværk. For det andet er neurale netværk meget generelle, det netværk og den tilhørende oplæringsregel, der er anvendt i denne rapport, kunne – med meget små ændringer – ligeså godt have været anvendt til orddeling eller til forudsigelse af proteiners 3-dimensionale struktur og uden ændringer til at forudsige alle mulige andre tidsserier.

## Taksigelser

Jeg vil godt have lov at takke Søren E. Larsen, Afdelingen for Meteorologi og Vindenergi, Forskningscenter Risø for hans altid velvillige hjælp og støtte i forbindelse med dette projekt såvel som andre. Endvidere vil jeg gerne takke mine to vejledere fra mit speciale Benny Lautrup, Niels Bohr Institutet og Aksel Walløe Hansen, Geofysisk Institut, begge Københavns Universitet.

# Referencer

- Andersen, A.H., 1984: *Statistik 1*. Afd. f. teor. stat., Matematisk Institut, Aarhus Universitet.
- Anderson, R.L., 1942: *Distribution of the Serial Correlation Coefficient*. Ann. Math. Stat. s 1. **13**.
- Ayres, F. Jr., 1962: *Theory and Problems of Matrices*. Schaum Publishing Company.
- Barnard, G. A., 1949: *Statistical Inference*. Jour. Royal Stat. Soc, **B11**, 116.
- Bartlett, M. S., 1946: *On the Theoretical Specification of Sampling Properties of Autocorrelated Time Series*. Jour. Royal Stat. Soc., **B8**, 27.
- Birnbaum, A., 1962: *On the Foundations of Statistical Inference*, Jour. Amer. Stat. Assoc., **57**, 269
- Bohr, H. et al., 1988: *Protein secondary structure and homology by neural networks - the  $\alpha$ -helices in rhodopsin*. FEBS Lett., **241** 223-228.
- Box, G. E. P. and G. M. Jenkins, 1962: *Some Statistical Aspects of Adaptive Optimization and Control*. Jour. Royal Stat. Soc., **B24**, 297.
- Box, G. E. P. and G. M. Jenkins, 1963: *Further Contributions to Adaptive Quality Control: Simultaneous Estimation of Dynamics: Non-zero Costs*. Bull. Intl. Stat. Inst., 34th Session. s 943. Ottawa, Canada.
- Box, G. E. P., G. M. Jenkins and D. W. Bacon, 1967: *Models for Forecasting Seasonal and Non-seasonal Time Series*. Advanced Seminar on Spectral Analysis of Time Series, ed. B. Harris. s 271. John Wiley, New York.
- Box, G. E. P. and G. M. Jenkins, 1968a: *Discrete Models for Feedback and Feed-forward Control*. The Future of Statistics, ed. D. G. Watts. s 201. Academic Press, New York.
- Box, G. E. P. and G. M. Jenkins, 1968b: *Some Recent Advances in Forecasting and Control, I*. Applied Stat., **17**, 91.
- Box, G. E. P. and G. M. Jenkins, 1969: *Discrete Models for Forecasting and Control*. Encyclopedia of Linguistics, Information and Control. s 162. Pergamon Press.
- Box, G. E. P. and G. M. Jenkins, 1970: *Time Series Analysis Forecasting and Control*. Holden-Day.
- Box, G. E. P. & D. A. Pierce, 1970: *Distribution of Residual Autocorrelations in Autoregressive-Integrated Moving Average Time Series Models*. Jour. Amer. Stat. Assoc., **65**. s 1509.
- Brunak, S. og B. Lautrup, 1988: *Neurale Netværk - computere med intuition*. Nysyn, Munksgaard.
- Farmer, J. D. and J. J. Sidorowich, 1988: *Exploiting Chaos to Predict the Future and Reduce Noise*. Los Alamos National Laboratory, Los Alamos. LA-UR-88-901.
- Fisher, R.A., 1956: *Statistical Methods and Scientific Inference*. Oliver and Boyd, Edinburgh. Grenander, U. and M. Rosenblatt, 1957: *Statistical Analysis of Stationary Time Series*. John Wiley, New York.
- Haken, H., 1975: *Analogy between higher instabilities in fluids and lasers*. Phys. Lett, **A53**, s. 77-78.
- Hald, A., 1952: *Statistical Theory with Engeneering Applications*. John Wiley, New York.
- Jahnsen, H. og O. Kiehn, 1989: *Nysyn på hjernens netværk*. Naturkampen nr. 51.
- Jenkins, G. M. and D. G. Watts, 1968: *Spectral Analysis and Its Applications*, Holden-Day, San Francisco.
- Kirkpatrick, S., C. D. Gelatt Jr. and M. P. Vecchi, 1983: *Optimization by simulated annealing*. Science, 220, 671-680.

- Krogh, A., 1989: *Neurale Netværk*. Gamma, 76.
- Lapedes, A. og R. Farber, 1987.: *Nonlinear Signal Processing Using Neural Networks: Prediction and System Modelling*. Preprint LA-UR-87-2662.
- Lautrup, B., 1989: *Lecture Notes on Feed-Forward Neural Networks*. Niels Bohr Institutet. Copenhagen, Denmark.
- Lippmann, R.P., 1987: *An introduction to Computing with Neural Nets*. IEEE ASSP Magazine, 4(2), 4-22.
- Lorenz, E. N., 1963.: *The Predictability of hydrodynamic flow*. Trans. N. Y. Acad. Sci., serie II, 25, 409-432.
- Lorenz, E. N., 1969: *Atmospheric Predictability as Revealed by Naturally Occuring Analogues*. J. Atmos. Sci., 25, 636-646.
- Makram-Ebeid, S., J.-A. Sirat and J.-R. Viala, 1989: *A Rationalized Error Back-propagation Learning Algorithm*. Proceedings from the IJCNN conference, Washington DC. June 18-22.
- Moody, J. 1989.: *Fast Learning in Multi-Resolution Hierarchies*. Research Report YALEU/ DSC/RR-681. Yale University.
- Moody, J. and C. Darken, 1989: *Fast Learning in Networks of Locally-Tuned Processing Units*. Neural Computation, 1, 281-94.
- Nielsen, N. Woetmann og A. W. Hansen, 1986: *Vejrets rytme på Fanø*. Vejret nr. 2, 24-26.
- Nielsen, N. Woetmann, 1988: *Nogle træk ved klimaet på Fanø 1875-1980*. Vejret nr. 2, 3-13.
- Noter til matematik A. Københavns Universitet, 1978.
- Owens and Filkin, 1989: *Efficient Training of the Back Propagation Network by Solving a System of Stiff Ordinary Differential Equations*. Proceedings from the IJCNN conference, Washington DC. June 18-22.
- Petersen, E.L. og S.E. Larsen, 1978: *A statistical study of a composite isotopic paleotemperature series from the last 700,000 years*. Tellus, 30, 193-200.
- Peterson, E. W. og S. E. Larsen, 1983: *Klimatiske ændringer over det sidste hundrede år ved Fanø*. Vejret nr. 3, 18-20.
- Peterson, E. W. og S. E. Larsen, 1984: *Analyse af vindobservationer fra Fanø i perioden 1872-1980*. Vejret nr. 1, 12-17.
- Quenouille, M.H, 1949: *Approximate Tests of Correlation in Time Series*. Jour. Royal Stat. Soc., B11, 68.
- Ruelle, D., F. Takens, 1971: *Nature of Turbulence*. Comm. Math. Phys., 20, s. 167.
- Rumelhart, D. E., J. E. McClelland and the PDP Research Group, 1986: *Parallel Distributed Processing. Explorations in the Microstructure of Cognition*. Volume 1: Foundations. MIT Press.
- Russell, D.A. et al., 1980: *Dimensions of strange attractors* Phys. Rev. Lett, 45, s. 1175-78.
- Scheid, F., 1968: *Numerical Analysis*. Schaum's Outline Series. McGraw-Hill Book Company. New York.
- Schuster, A., 1898: *On the investigation of hidden periodicities*. Terr. Mag. 3, 12.
- Schuster, H. G., 1984: *Deterministic Chaos. An Introduction*. Physik-Verlag. Fed. Rep. of Germany.
- Shustorovich, A., 1989: *A Neural Model of Interpolation or Interpolation with Blobs*. Advanced Computing Laboratory, Eastman Kodak Company. Poster ved IJCNN konferencen, Washington D. C., 18-22 juli.
- Smagorinsky, J., 1969: *Problems and Promises of Deterministic Extended Range Forecasting*. Bull. Amer. Meteor. Soc., 50, 286-313.
- Stork, D. G. and J. Hall, 1989: *Is Back-propagation Biologically Plausible*. Proceedings from the IJCNN conference, Washington DC. June 18-22.
- Swinney, H. et al., 1978: *Transition to turbulence*. Physics Today, 31(8), s. 41.



- Tomita, K. et al., 1979: *Chaotic response of a limit cycle*. J. Stat. Phys., **21**, s. 65.
- Walker, G, 1931: *On Periodicity in Series of Related Terms*. Proc. Royal Soc., **A131**, 518.
- Widrow, R. J. & M. E. Hoff, 1960: *Adaptive Switching Circuits*. Inst. of Radio Engineers, Western Electronic Show and Convention, Convention Record, Part 4, 96-104.
- Yaglom, A. M., 1955: *The Correlation Theory of Processes Whose  $n$ th Difference Constitute a Stationary Process*. Matem. Sb., **37** (79), 141.
- Yule, G.U, 1927: *On a Method of Investigating Periodicities in Disturbed Series, with Special Reference to Wölfer's Sunspot Numbers*. Phil. Trans., **A226**, 267.





**Bibliographic Data Sheet****Risø-R-656(DA)**

---

**Title and author(s)**

Neural networks utilized for the prediction of a geophysical time series  
A comparison

Lars Landberg

---

**ISBN**

87-550-1859-9

---

**ISSN**

0106-2840

---

**Dept. or group**

Meteorology and Wind Energy

---

**Date**

December 1992

---

**Groups own reg. number(s)**

---

**Project/contract No(s)**

---

**Pages**

76

---

**Tables**

7

---

**Illustrations**

40

---

**References**

54

---

**Abstract (Max. 2000 char.)**

Two methods of predicting a geophysical time series are described and evaluated on a temperature series measured at Fanø, Denmark, during a period of 94 years. The first method involves the use of neural networks (i.e. large arrays of densely connected simple computational units), and the second the ARIMA-process (Auto Regressive Integrated Moving Average). The neural networks, the ARIMA-model, and the presistence model all give the same result for a one day forecast. For a five day forecast neural networks perform better than both the ARIMA-model and persistence.

---

**Descriptors INIS/EDB**

---

**Available on request from:**

Risø Library, Risø National Laboratory (Risø Bibliotek, Forskningscenter Risø)

P.O. Box 49, DK-4000 Roskilde, Denmark

Phone (+45) 42 37 12 12, ext. 2268/2269 · Telex 43 116 · Telefax (+45) 46 75 56 27